

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

LARGE-EDDY SIMULATION OF A TURBULENT MIXING LAYER

by

N. N. MANSOUR,

J. H. FERZIGER,

and

W. C. Reynolds

(NASA-CR-156575) LARGE-EDDY SIMULATION OF A
TURBULENT MIXING LAYER (Stanford Univ.)
210 p HC A10/MF A01 CSCL 01A

N78-22027

Unclas

G3/02

16615

Prepared from work done under Grant

NASA-NgR-05-020-622



Report No. TF-11

Thermosciences Division
Department of Mechanical Engineering
Stanford University
Stanford, California

April 1978

LARGE-EDDY SIMULATION OF A TURBULENT MIXING LAYER

by

N. N. Mansour, J. H. Ferziger, and W. C. Reynolds

Prepared from work done under Grant
NASA-Ngr-05-020-622

Technical Report Number TF-11

Thermosciences Division
Department of Mechanical Engineering
Stanford University
Stanford, California

April 1978

Acknowledgments

The authors gratefully acknowledge the many useful discussions and contributions of Mr. P. Moin. We are indebted to Dr. U. B. Mehta for helping get this work started and familiarizing the senior author with the computation facility at NASA-Ames Research Center. Dr. A. Leonard made numerous suggestions and corrections to the manuscript which the authors gratefully acknowledge.

The senior author is indebted to the National Council for Scientific Research (Lebanon) for its support at the initial stages of his graduate studies.

The diligent efforts and editing of Ms. Ruth Korb resulted in a superb job of preparation of the final copy. This work was supported by NASA-Ames Research Center under Grant NASA-NgR-05-020-622.

Abstract

The three-dimensional, time-dependent (incompressible) vorticity equations have been used to simulate numerically the decay of isotropic box turbulence and time-developing mixing layers. The vorticity equations are spatially filtered to define the large-scale turbulence field, and the subgrid scale turbulence is modeled. A general method has been developed to show numerical conservation of momentum, vorticity, and energy that is much simpler than previous methods and is widely applicable. The terms that arise from filtering the equations have been treated (for both periodic boundary conditions and no-stress boundary conditions) in a fast and accurate way by using fast Fourier transforms. Use of vorticity as the principal variable is shown to produce results equivalent to those obtained by use of the primitive variable equations.

A new subgrid scale model is used in conjunction with the vorticity equations and is shown to produce results that compare well with the experimental results. The new model offers advantages both in computational speed and in storage.

The vortex-pairing mechanism, observed in the spatially developing counterpart of the time-developing mixing layer, has been simulated numerically. It is interesting to note that with simply two vortices pairing, self-similar mean velocity and mean turbulence intensity profiles are obtained. The vortex-pairing mechanism is shown to be persistent even with the presence of large-amplitude, three-dimensional background turbulence. A number of different initial fields have been studied. The presence of large organized structures, in the initial conditions, is shown to be essential in order to predict growth rates of the mixing layers comparable to those observed experimentally. The rate of growth is found to be very dependent on the initial field, a fact also observed experimentally.

Table of Contents

	Page
Acknowledgments	iii
Abstract	iv
List of Tables	vii
List of Figures	viii
Nomenclature	xiii
 Chapter	
1 INTRODUCTION	1
1.1 Background	1
1.2 Experimental Background	3
1.3 Motivation and Objectives	5
1.4 Overview	6
2 THEORETICAL FOUNDATIONS	8
2.1 Definitions of the Large and Subgrid Scales	8
2.2 Dynamical Equations in Vorticity Form	9
2.3 Subgrid Scale Models	10
2.3.1 Model ω -1	11
2.3.2 Model ω -2	12
2.4 Filtering	13
2.4.1 Sharp Cut-off (SCK) Filter	13
2.4.2 Gaussian (GS) Filter	14
2.5 Computing Velocity Field from the Vorticity Field	14
2.6 Summary	15
3 NUMERICAL METHOD	17
3.1 Notations	17
3.2 Numerical Approximation	17
3.2.1 Fourth-Order Scheme	18
3.2.2 Pseudo-Spectral Method	20
3.3 Time Differencing	25
3.4 Conservation Properties	25
3.5 Differenced Vorticity Equations	29
3.6 Poisson Equation	30
3.7 Numerical Filtering	31
3.8 Summary	34
4 DECAY OF ISOTROPIC TURBULENCE	36
4.1 Background	36
4.2 Initial Conditions	36
4.3 Selection of C_v	37
4.4 Results	37
4.4.1 Fourth-Order Finite Differences	38
4.4.2 Pseudo-Spectral Method	38
4.5 Computational Details	39

Chapter		Page
5	MIXING LAYER: TWO-DIMENSIONAL COMPUTATIONS	41
	5.1 Preview	41
	5.2 Some Experimental Results	41
	5.3 Boundary Conditions	44
	5.4 Initial Conditions	45
	5.5 Mesh Size Selection	46
	5.6 Selection of β	48
	5.7 Mean Velocity Profiles	48
	5.8 Mean Turbulence Intensity Profiles	49
	5.9 Summary	49
6	MIXING LAYER: THREE-DIMENSIONAL COMPUTATIONS	51
	6.1 Preview	51
	6.2 Boundary Conditions and Mesh-Size Selection	51
	6.3 Initial Conditions	52
	6.4 Momentum Thickness	53
	6.5 Mean Velocity Profiles	54
	6.6 Mean Turbulence Intensity Profiles	54
	6.7 Vorticity Contours	56
	6.8 Two-Point Correlations	57
	6.9 Summary and Conclusions	59
7	CONCLUSIONS AND RECOMMENDATIONS	60
	References	63
Appendix		
A	Subgrid Scale Models for the Vorticity Equations	136
B	Fast Discrete Sine Transform (FDST); Fast Discrete Cosine Transform (FDCT)	138
C	Effect of a Sinusoidal Vorticity Perturbation on a Uniform Vortex Array	140
	1. Initial Conditions	140
	2. Results	140
D	Interactions Between Streamwise and Spanwise Vorticity	142
	1. Initial Conditions	142
	2. Results	142
E	Computer Program Written to Calculate Turbulent Mixing Layers	160

List of Tables

Table	Page
1.1 Experimental Results	67
4.1 Computations of the Decay of Isotropic Turbulence	75
6.1 Three-Dimensional Computations of Turbulent Mixing Layers .	103

List of Figures

Figure	Page
1.1 r.m.s. streamwise velocity profiles for different initial conditions (experimental results from Foss, 1977)	69
1.2 Mean velocity profiles for different initial conditions (experimental results from Foss, 1977).	70
2.1 Filtered point vortex with SCK (sharp cut-off in k-space) filter	71
2.2 Filtered point vortex with a GS (Gaussian in real space) filter	72
3.1 Comparison of modified wave number	73
3.2 Filtered top-hat function	74
4.1 Decay of mean square filtered velocity for $16 \times 16 \times 16$ mesh. $\langle \rangle$ = average over all space	76
4.2 Filtered energy spectra. Fourth-order differencing with 16^3 mesh; model ω -1, $C_v = 0.235$	77
4.3 Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -1, $C_v = 0.212$	78
4.4 Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -1, $C_v = 0.213$	79
4.5 Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -2. $C_v = 0.186$	80
4.6 Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -2. $C_v = 0.188$	81
4.7 Filtered energy spectra. Pseudo-spectral computation with 32^3 mesh, 2nd-order differencing for model ω -2. $C_v = 0.188$	82
5.1 Coherent structure in a mixing layer (Roshko, 1976). Dashed box: schematic of a computational box that moves approximately with the mean velocity	83
5.2 Mixing layer. Experimental setup and coordinate system	84
5.3 Computational box and coordinate system	85
5.4 Initial conditions setup $\left(\beta = \frac{1}{2} - \frac{(x_1 - x_2)}{L} \right)$	86
5.5 Non-dimensional momentum thickness (θ/θ_{in}) as a function of time for various β . Two-dimensional computations	87

List of Figures (cont.)

Figure		Page
5.6a	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 6/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = .702$)	88
5.6b	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 5/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.564$)	89
5.6c	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 4/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.444$)	90
5.6d	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 3/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\bar{\omega}_{2,\max} = 0.421$)	91
5.6e	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 2/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.416$)	92
5.6f	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 1/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.415$)	93
5.7a	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 6/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.394$)	94
5.7b	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 5/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.358$)	95
3.7c	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 4/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.322$)	96
5.7d	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 3/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.276$)	97

List of Figures (cont.)

Figure		Page
5.7e	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 2/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,max} = 0.248$)	98
5.7f	Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 1/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,max} = 0.245$)	99
5.8	Mean velocity profiles. Two-dimensional computations ($\beta = 3/16$)	100
5.9	Mean turbulence intensity profiles. Two-dimensional computations ($\beta = 3/16$)	101
6.1	Three-dimensional computation box. Random velocity setup and coordinate system	102
6.2	Non-dimensional momentum thickness (θ/θ_{in}) as a function of time. Three-dimensional computations	103
6.3	Non-dimensional momentum thickness (θ/θ_{in}) as a function of time	104
6.4a	Mean velocity profiles. Three-dimensional computation (case a)	105
6.4b	Mean velocity profiles. Three dimensional computation (case b)	106
6.4c	Mean velocity profiles. Three-dimensional computations (case c)	107
6.5	Integral of the turbulence energy as a function of time .	108
6.6a	Mean turbulence intensity profiles. Three-dimensional computations (case a)	109
6.6b	Mean turbulence intensity profiles. Three-dimensional computations (case b)	110
6.6c	Mean turbulence intensity profiles. Three-dimensional computations (case c)	111
6.7	Mean turbulence intensity profiles. Three-dimensional computation ($C_v = 0$)	112
6.8	Integral of the turbulence energy as a function of time .	113

List of Figures (cont.)

Figure	Page
6.9a Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\bar{\omega}_{2,\max} = 0.228$)	114
6.9b Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x-z planes, at time $T = 0$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels	115
6.9c Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x-z planes, at time $T = 0$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels	119
6.10a Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\bar{\omega}_{2,\max} = 0.185$)	123
6.10b Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x-z planes, at time $T = 16.78$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels	124
6.10c Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x-z planes, at time $T = 16.78$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels	128
6.11a Two point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case a)	132
6.11b Two-point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case b)	133
6.11c Two-point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case c)	134
7.1 Formation of streamwise cellular structures in a mixing layer	135
C.1 Non-dimensional momentum thickness (θ/θ_{in}) as a function of time for various C_2/C_1	145
C.2 Mean velocity profiles ($C_2/C_1 = 0.1$)	146
C.3 Mean turbulence intensity profiles ($C_2/C_1 = 0.1$)	147

List of Figures (cont.)

Figure	Page
D.1 Contour plots of the streamwise vorticity ($\bar{\omega}_1$) at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels	148
D.2 Non-dimensional momentum thickness (θ/θ_{in}) as a function of time	149
D.3a Mean velocity profiles (case a)	150
D.3b Mean velocity profiles (case b)	151
D.4a Mean turbulence intensity profiles (case a)	152
D.4b Mean turbulence intensity profiles (case b)	153
D.5a Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case a)	154
D.5b Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case b)	155
D.6a Contour plots of the streamwise vorticity ($\bar{\omega}_1$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels (case a)	156
D.6b Contour plots of the streamwise vorticity ($\bar{\omega}_1$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case b)	157
D.7 Projection of the vorticity vector in an x-z plane ($y/\theta_{in} = 4.09$) at time $T = 16.78$	158
D.8 Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in a y-z plane ($x/\theta_{in} = 9.55$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\omega_{2,max} = 0.714$)	159

Nomenclature

A	Integral of the Gaussian filter
A_i	Integral of the Gaussian filter in the i -direction
C_v	Subgrid scale constant
$\overline{E}(k)$	Filtered energy spectrum
f, g	Flow variables
$\overline{f}, \overline{g}$	Filtered (large-scale) components of flow variables
f', g'	Subgrid scale components of flow variables
$G(x)$	Filter function
$\hat{G}(k)$	Fourier transform of the filter function
$G_D(x)$	Discretized filter in real space
$\hat{G}_D(k)$	Discretized filter in k -space.
h	Mesh size in any given direction
h_i	Mesh size in the i -direction
k	Wave number
k_i	Wave number in the i -direction
k'_i	Modified wave number in the i -direction
k_c	Cut-off wave number
$\mathcal{L}$	Length scale of large eddies
L	Length of computational box in any given direction
L_i	Length of computational box in the i -direction
M	Size of the experimental turbulence-generating grid
N	Number of mesh points in any given direction
N_i	Number of mesh points in the i -direction
q	r.m.s. velocity
$r \equiv u_2/u_1$	velocity ratio

R_T	=	$q\lambda/\nu$, Reynolds number based on large eddy length
R_λ	=	$q\lambda/\nu$, Reynolds number based on Taylor micro-scale
S		Scalar
S_{ij}	$\equiv$	$\frac{1}{2} \frac{\partial}{\partial x_j} \bar{u}_i + \frac{\partial}{\partial x_i} \bar{u}_j$, strain rate tensor
T	=	Non-dimensional time
	$\equiv$	$\frac{U_o t}{M}$, decay of isotropic turbulence-
	$\equiv$	$\frac{\Delta u t}{\theta_{in}}$, mixing layer
t		Real time
u_i		Velocity in the i-direction
$\bar{u}_i$		Filtered velocity in the i-direction
u'_i		Subgrid scale velocity in the i-direction
u_1		Velocity of high-speed side
u_2		Velocity of low-speed side
V		General vector
V_i		Component of a general vector in the i-direction
W_{ij}		Curl of subgrid scale stress
x		Streamwise coordinate
x_o		Virtual origin of a mixing layer
y		spanwise coordinate
z		Cross-flow coordinate

Greek letters

Δ		Filter width (= 2h) in any direction
Δ_i		Filter width in the i-direction
Δu	=	$u_1 - u_2$ velocity difference
ϵ		Total energy dissipation
ϵ_{ijk}		The completely anti-symmetric tensor of rank 3

η	$z/(x-x_0)$ self-similarity coordinate
γ	Constant (= 6) in Gaussian filter
λ	Taylor microscale
ν_T	Eddy viscosity
$\omega_i \equiv$	$\epsilon_{ipq} \frac{\partial}{\partial x_p} u_q$, vorticity component in the i-direction
$\bar{\omega}_i$	Filtered vorticity component in the i-direction
ω'_i	Subgrid scale vorticity component in the i-direction
ψ_i	Component of vector potential in the i-direction
ρ	Density
$\tau_{ij} \equiv$	$\overline{u'_i u'_j} + \overline{u'_j u'_i} + \overline{u'_i u'_j} - \frac{1}{3} (\overline{u'_k u'_k} + 2\overline{u'_k u'_k}) \delta_{ij}$
σ	Spread parameter
σ_0	Spread parameter for $r = 0$
θ	Momentum thickness
$\langle \rangle_{xy}$	Horizontal planar average

Arguments

(i,j,k) Computational mesh index for (x,y,z)

Superscripts

(n) Time step

Chapter 1

INTRODUCTION

1.1 Background

Turbulent flows have been the subject of experimental and theoretical investigations since the last century. Despite the formidable amount of effort invested in this field, our ability to predict flows of technical importance remains severely limited. The major difficulty encountered by the theoretical investigations arises from the nonlinear character of the equations of motion. Statistical averages of the equations of motion give rise to the so-called Reynolds stresses. The equations for the Reynolds stresses in turn give rise to higher-order statistical quantities, and so on. The usual approach to computing turbulent flows is to model the terms that arise from the nonlinear character of the equations of motion. This approach usually requires a great deal of experimental information.

We know the underlying physical principles of most fluid flows, and the quantities of interest are completely determined by known equations. With the introduction of large computers, three-dimensional, time-dependent computation of turbulent flows has become possible. However, in order to resolve all the scales of motion even in the simplest turbulent flow, namely, the isotropic homogeneous case, Kwak et al. (1975) estimated the number of mesh points needed in any given direction to be

$$N = R_T^{3/4} \quad (1.1)$$

where

$$R_T = (q\mathcal{L}/\nu),$$

ν = kinematic viscosity,

$\mathcal{L}$ = length scale of large eddies, and

q = r.m.s. velocity.

Equation (1.1) shows that one can do a full simulation only at very low Reynolds number. Indeed, Clark et al. (1977), using a $64 \times 64 \times 64$ mesh system, were able to solve the isotropic homogeneous turbulence

problem for $R_\lambda = q\lambda/\nu = 38.1$, where λ is the Taylor microscale. Their predicted results compared well with the experimental results. However, turbulent flows of technical importance have much higher Reynolds numbers, and all the scales of motion cannot be resolved for these flows.

One of the more promising approaches to solving turbulence problems is "large-eddy simulation". In large-eddy simulation, one calculates the large-scale turbulent motions with a relatively coarse time-dependent, three-dimensional computation that uses some sort of model (the "subgrid scale model") for the small scales. The basic motivations for this approach are twofold. First, experimental observations of turbulent flows show that the large turbulent structures differ markedly from one flow type to another (e.g., jet vs. boundary layer), but the small-scale turbulent structures are quite similar. Thus, while there is little hope of concocting a "universal" model for the large structures, it may be possible to do so for the small-scale motions. Second, as computer capabilities grow, our capability of resolving smaller scales will grow and the effects of the subgrid scale model will diminish. Thus, while we are limited to simple flows with the present computer capabilities, large-eddy simulation is a tool that may be used on future generation computers.

Kwak et al. (1975) and Shaanan et al. (1975) have shown that homogeneous turbulent flows can be simulated reasonably well with a relatively small number of mesh points ($16 \times 16 \times 16$). Orszag and Pao (1974), using a $32 \times 32 \times 32$ mesh system, predicted the momentumless wake of a self-propelled body. Deardorff (1970) and Schuman (1973) computed the central region of a plane channel flow using the large-eddy simulation approach. While Deardorff and Schumann did not handle the wall (no slip) problem, Moin et al. (1978) have solved the channel flow problem, including the laminar sublayer. In this work we shall study the time-developing, two-stream mixing layer.

Previous works on prediction of the two-stream mixing layer have concentrated on the initial stages (roll-up) of the development of the layer. Patnaik et al. (1976), starting with an initial distribution that is an unstable eigensolution of the Taylor-Goldstein equation, predicted the two-dimensional roll-up of a stably stratified horizontal mixing layer. Another method that has been used to compute the mixing layer in two

dimensions is the vortex-tracing method used by Ashurst (1977). This method suffers from high computational costs and ad-hoc assumptions concerning the effects of viscosity. The high computational cost of the vortex-tracing method can be reduced by using the vortex-in-cell method (Wang, 1977). These works have treated two-dimensional cases, but the mixing layer exhibits three-dimensionality. This is apparent from the shadowgraph pictures of Brown and Roshko (1974) and the spanwise velocity fluctuation measurements of Spencer and Jones (1971).

1.2 Experimental Background

The two-dimensional turbulent mixing layer plays an essential role in many technological problems. For example, the initial regions of planar jets can be approximated as two independent, two-dimensional mixing layers. Flow over a backward-facing step (with a large step height) is another example of the two-dimensional mixing layer. Many other flow situations can be identified with the mixing layer. In combustion processes, fluid mechanics plays a major role in mixing the reactants, and better understanding of turbulent mixing is needed. The mixing layer is perhaps the simplest situation in which two flows come into contact; obviously the ability to analyze simple problems is necessary before one can analyze more complicated ones.

In 1947, Liepmann and Laufer studied the mixing layer and established the general features of the flow. However, the fundamental understanding of the structure of the flow is still far from complete, and many controversial questions need to be answered. We shall address some of these questions. The reader is referred to Murthy (1975) for an extensive review and interpretation of the available literature on the mixing layer. With the advancement of the techniques of hot-wire anemometry, Wygnanski and Fiedler (1970) attempted to reproduce Liepmann and Laufer data and extend it to include other measurements. However, differences in intensity levels and rate of growth of the layer emerged. These differences were attributed to the presence of a trip wire in the Wygnanski and Fiedler experiment that was not used by Liepmann and Laufer. Batt (1975) studied both configurations and showed that the differences are due to the tripping of the layer. Foss (1977) investigated the effects of the

laminar/turbulent boundary layer states on the development of a plane mixing layer. He found that the development of the layer is dependent on the initial conditions (the status of the boundary layers before the two streams merge). Figs. 1.1 and 1.2 show r.m.s. fluctuations of the streamwise velocity and the mean velocity profiles obtained by Foss. These figures show that different self-similar stages are obtained for different initial conditions. Foss argues that this is due to the sensitivity of choosing the virtual origin of the mixing layer (x_0) and that the character of the (initial) disturbance, not its amplitude, is responsible for the substantial effect on the virtual origin. More recently, Oster et al. (1977) showed that by oscillating the initial conditions of the mixing layer they can more than double the growth rate of the layer. The effect depends on the frequency and amplitude of the oscillations introduced. These experimental results show that the "universality" of the self-similar stage of the mixing layer is in doubt, at least up to $Re = 1.5 \times 10^6$. Fiedler and Thies (1977) showed that the two-dimensional shear layer only slowly reaches a self-similar state and that every disturbance is of long influence. Table 1.1 shows tabulated results extracted from the Fiedler and Thies paper, and it can be clearly seen that different experiments predict different growth rate of the layer.

Winant and Browand (1974), using dye visualization in a mixing layer, observed that initially the fluid rolls up into discrete, two-dimensional vortical structures. These structures then interact by rolling around each other to form a single larger structure. This pairing process controls the growth of their mixing layer. Brown and Roshko (1974) also observed the amalgamation process at Reynolds number 2.5×10^5 . Chandrsuda and Bradshaw (1975) argue that the two-dimensional, large-eddy structure observed by Brown and Roshko is unlikely to survive indefinitely if the ambient entrained fluid is weakly turbulent. They advance the argument: "It is probable that if the Brown and Roshko type of orderly structure is once formed it can last for a large number of characteristic wavelengths -- that is, up to high Reynolds numbers based on streamwise distance -- but not indefinitely. The question can be settled only by measurements in a two-stream mixing layer at a much higher Reynolds number than was used by Brown and Roshko." Dimotakis and Brown (1976) showed the existence of

large structures at Reynolds number $= 3 \times 10^6$ and attributed the growth of the mixing layer to both pairing and "tearing". Tearing is described in their paper as an event where "a large structure will occasionally find itself in the vicinity of another, or in between two others, in whose straining field it disintegrates." The tearing process was first advanced by Moore and Saffman (1975) on the basis of exact solutions for uniform vortices in straining fields.

1.3 Motivation and Objectives

In many flows of practical interest there are interactions between irrotational regions and turbulent regions. Examples of such flows are the shear layer, turbulent jets, and turbulent boundary layers with irrotational free stream flow. In such flows, the regions are separated by a very thin superlayer across which there is normally a jump in the vorticity components parallel to the layer. The dynamical equations for the vorticity seem to be suited to simulate such flows, since the vorticity is identically zero in the irrotational region. However, previous workers used the dynamical equations in the primitive variables (velocity, pressure) and there has been doubt (Orszag and Israeli, 1974) that the vorticity equations could be used to solve turbulent flow problems. Our objectives were therefore as follows:

- To explore the feasibility of using the vorticity equation to simulate turbulent flows.
- To find a subgrid scale model appropriate to the vorticity equations and to determine any constants in this model.
- To simulate a turbulent flow with interactions between turbulent regions and non-turbulent irrotational regions; we chose the mixing layer.

In order to use the three-dimensional, time-dependent vorticity equations, we need to develop a numerical approximation based on these equations that conserves mass, momentum, vorticity, and energy. We also need to assess numerical finite-difference methods and, in particular, the fourth-order and pseudo-spectral methods.

1.4 Overview

The equations of motion of the large eddies are derived by averaging (filtering) the vorticity equations in space. In Chapter 2, we describe the approach to solving turbulent flow problems that is called large-eddy simulation. We show that the use of a filter that is smooth in the real space is required to handle rotational-irrotational regions. We present a new subgrid scale model to be used in conjunction with the vorticity equations that is much simpler and faster than the one that would be obtained from the more commonly used Smagorinsky model.

In Chapter 3, we describe the numerical methods used in this work, briefly discussing the fourth-order and pseudo-spectral approximations and numerical filtering. We develop a numerical approximation to the vorticity equation that conserves mass, momentum, vorticity, and energy, and a method of deriving conservation properties that is much simpler than previous methods and is widely applicable. We present a new treatment of the filtered convective and stretching terms that is more accurate and faster than previously used methods.

In Chapter 4, the isotropic homogeneous turbulence problem is solved using both fourth-order differencing and the pseudo-spectral approximation. The numerical approximations to the partial derivatives of the subgrid scale model are discussed. We show that the use of the vorticity equation to solve turbulent flow problems is feasible and that the new model produces results equivalent to those produced by previously established models.

In Chapter 5, we discuss the two-dimensional computation of a mixing layer. An array of vortices is perturbed, and the momentum thickness growth rate is discussed as a function of the perturbation. It is interesting to note that self-similar, mean velocity and turbulence intensity profiles are obtained with vortex pairing.

In Chapter 6 a three-dimensional computation of a turbulent mixing layer is studied. It is found that the presence of large structures in the initial conditions is essential for the successful prediction of turbulent mixing layers. Our studies of different initial conditions produce different growth rates of the layer -- a fact supported experimentally. Self-similar, mean-velocity profiles are obtained with different flow structures. However, turbulence intensity profiles show a rapid decay when

large turbulent structures are not present. We show that our subgrid scale model inhibits the production of turbulent fluctuations when we start with random turbulent fluctuations added to a mean velocity profile, i.e., the model is incapable of handling transitional flows, with present computational limitations.

Chapter 2

THEORETICAL FOUNDATIONS

2.1 Definitions of the Large and Subgrid Scales

In the previous chapter it was shown that, due to computer limitations, one cannot do a full simulation of the dynamical equations of turbulent fluid motion except at extremely low Reynolds numbers. We pointed out that the large-scale turbulent structures differ markedly from one flow to another (e.g., jet vs. boundary layer), while the small-scale turbulent structures are quite similar, and that large-eddy simulation is a promising approach.

In the large-eddy simulation approach, the first and most fundamental step is defining the large-scale field. A general approach that recognizes the continuous nature of the flow variables is the "filter function" approach of Leonard (1973). If f is some flow variable, we can decompose it as follows:

$$f = \bar{f} + f' \quad (2.1)$$

where $\bar{f}$ is the large-scale (filtered) component and f' is the residual field. Leonard defined the filtered field by:

$$\bar{f} = \int_{\Omega} G(\underline{x}-\underline{x}') f(\underline{x}') d\underline{x}' \quad (2.2)$$

where $G(\underline{x}-\underline{x}')$ is the filter function, and the integral is extended over the whole flow field. One can think of $\bar{f}$ as a local spatial-averaged field.

It can be shown that if G is piecewise continuously differentiable and $G(\underline{r})$ goes to zero as $r \rightarrow \infty$ and is integrable over an infinite domain, then

$$\overline{\frac{\partial f}{\partial x}} = \frac{\partial \bar{f}}{\partial x} \quad (2.3a)$$

However, in general,

$$\overline{\overline{fg}} \neq \overline{\overline{f}} \overline{\overline{g}} \quad (2.3b)$$

Properties (2.3a) and (2.3b) will be used in deriving the dynamical equations of the large scales motion.

2.2 Dynamical Equations in Vorticity Form

In Chapter 1, we pointed out that in many flows of practical interest there are interactions between irrotational regions and rotational turbulent regions. Examples of such flows are shear layers, turbulent jets, and turbulent boundary layers with irrotational free streams. In such flows the regions are separated by a very thin superlayer across which there is normally a jump in the vorticity parallel to the layer. These flows are a challenge to the experimentalist; the difficulties arise from the fact that it is hard to determine the region of the flow in which the measurements are made. One faces a similar problem in trying to simulate such flows numerically. The difficulty arises from the fact discussed earlier, that it is impossible to capture all of the scales of motion in the turbulent region. The best we can do is to filter the dynamical equations to obtain equations that describe the behavior of the large eddies, and to model the small scales. Since in the irrotational region the vorticity is identically zero, the dynamical equations for the vorticity seem to be suited to simulate such flows.

Now let us derive the dynamical equations for the large-scale vorticity field. For an incompressible fluid with constant viscosity, the equations of motion for the primitive variables may be written:

$$\frac{\partial u_i}{\partial t} - \epsilon_{ijk} u_j \omega_k = - \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} - \frac{1}{2} u_i u_i \right) - \nu \epsilon_{ijk} \frac{\partial}{\partial x_j} \omega_k \quad (2.4)$$

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (2.5)$$

The vorticity equation is obtained by taking the curl of Eqn. (2.4). Operating on it with $\epsilon_{pqi} \partial / \partial x_q$ gives:

$$\frac{\partial}{\partial t} \omega_i + \frac{\partial}{\partial x_j} (u_j \omega_i - u_i \omega_j) = \nu \frac{\partial^2 \omega_i}{\partial x_j \partial x_j} \quad (2.6)$$

Multiplying Eqn. (2.6) by a filtering function $G(\underline{x}-\underline{x}')$ and integrating over the whole flow field, we obtain:

$$\frac{\partial}{\partial t} \bar{\omega}_i + \frac{\partial}{\partial x_j} (\overline{u_j \omega_i - u_i \omega_j}) = \nu \frac{\partial^2 \bar{\omega}_i}{\partial x_j \partial x_j} \quad (2.7)$$

The fact that a finite-difference approximation of Eqn. (2.7) would involve approximating higher derivatives of the velocity than would be the case with the primitive equations (Orszag and Israeli, 1974) need not worry us in this case. Since the equations are filtered, we shall be dealing with smooth functions.

As can be expected, when averaging nonlinear equations, we run into the closure problem; i.e., we need to express the quantities $\overline{u_j \omega_i}$ and $\overline{u_i \omega_j}$ in terms of $\bar{u}$ and $\bar{\omega}$. Expanding u and ω as in Eqn. (2.1), one obtains

$$\overline{u_j \omega_i - u_i \omega_j} = \overline{\bar{u}_j \bar{\omega}_i - \bar{u}_i \bar{\omega}_j} + W_{ij} \quad (2.8)$$

where

$$W_{ij} = \overline{u_j \omega_i' + u_j' \bar{\omega}_i - \bar{u}_i \omega_j' - u_i' \bar{\omega}_j + u_j' \omega_i' - u_i' \omega_j'} \quad (2.9)$$

We note that W_{ij} contains subgrid scale quantities and hence must be modeled.

2.3 Subgrid Scale Models

We first note that the model of W_{ij} should satisfy the following necessary conditions:

1. Antisymmetry, since W_{ij} is an antisymmetric tensor and therefore

$$\frac{\partial}{\partial x_i} \frac{\partial}{\partial x_j} W_{ij} = 0 \quad (2.10)$$

It is important to preserve the antisymmetry property of W_{ij} in order to assure $\partial \omega_i / \partial x_i = 0$, since the dynamical equations for the

vorticity do not contain a pressure-like term which could be used to adjust the divergence of the vorticity.

2. It should vanish in an irrotational region, since W_{ij} vanishes in such regions.
3. It should be an energy sink, since it represents subgrid-scale effects.

2.3.1 Model $\omega - 1$

Previous workers (Kwak et al., 1975; Shaanan et al., 1975), working with the filtered dynamical equations in the primitive variables, used an eddy-viscosity model for their subgrid-scale model. They modeled the term:

$$\tau_{ij} = \overline{u'_i u'_j} + \overline{u'_j \overline{u}_i} + \overline{u'_i \overline{u}_j} - \frac{1}{3} (\overline{u'_k u'_k} + 2\overline{u'_k \overline{u}_k}) \delta_{ij}$$

by setting

$$\tau_{ij} = -2\nu_T \overline{S}_{ij} \quad (2.11)$$

where

$$\overline{S}_{ij} = \frac{1}{2} \left(\frac{\partial}{\partial x_i} \overline{u}_j + \frac{\partial}{\partial x_j} \overline{u}_i \right) \quad (2.12)$$

is the strain rate tensor of the filtered field and ν_T is an eddy viscosity associated with the subgrid scale motions.

Smagorinsky (1963) suggested a model for ν_T

$$\nu_T = (C_S \Delta)^2 (2\overline{S}_{ij} \overline{S}_{ij})^{1/2} \quad (2.13)$$

where C_S is a constant and Δ is the filter width. We note that in a non-turbulent region this model of ν_T may have a non-zero value, and hence it may give rise to residual stresses. Since our main objective is to handle interactions between a turbulent region and a non-turbulent region, this model was rejected for the present work.

One way to avoid this difficulty is to relate ν_T directly to vorticity. Previous workers (Kwak et al., 1975; Donaldson, 1972) used

$$\nu_T = (C_V \Delta)^2 (\overline{\omega}_i \overline{\omega}_i)^{1/2} \quad (2.14)$$

where C_v is a constant. Clark et al. (1977) have shown that this model is as accurate as Smagorinsky's for homogeneous isotropic turbulence.

The dynamical equations for large-scale vorticity field could have been derived by taking the curl of the filtered dynamical equations for the primitive variables. Hence the curl of Eqn. (2.11) could be used to model W_{ij} ; this would give

$$W_{ij} = - \epsilon_{ijk} \frac{\partial}{\partial x_k} (2v_T \bar{S}_{kl}) \quad (2.15)$$

where $\bar{S}_{kl}$ and v_T are defined by Eqns. (2.12) and (2.14), respectively. We shall refer to this as Model ω -1.

2.3.2 Model ω -2

We note that the model given by Eqn. (2.15) involves computing the strain-rate tensor $\bar{S}_{kl}$, which is an expensive process. It also uses the velocity field and hence requires storage space for the velocity fields even after the convective and stretching terms have been computed. Much computational saving could be obtained with a model that involves only the vorticity field; one such model is

$$W_{ij} = - \frac{\partial}{\partial x_j} (v_T \bar{\omega}_i) + \frac{\partial}{\partial x_i} (v_T \bar{\omega}_j) \quad (2.16)$$

where v_T is defined by Eqn. (2.14). We shall refer to this as model ω -2.

Both models ω -1 and ω -2 can be shown to satisfy all three properties mentioned previously (see Appendix A). Model ω -2 offers computational as well as storage advantages over model ω -1 and will be tested in Chapter 4 (along with model ω -1), for the case of isotropic homogeneous turbulence.

2.4 Filtering

2.4.1 Sharp Cut-off (SCK) Filter

Analytically, a filter that divides the large scales and the subgrid scales into two distinct regions in the Fourier sense would be convenient. Then $\bar{f}$ would contain all scales larger than a cut-off scale, and the subgrid scales (f') would contain all scales smaller than this cut-off scale. A one-dimensional version of such a filter is

$$G(x-x') = \frac{\sin[k_c(x-x')]}{\pi(x-x')} \quad (2.17)$$

and its Fourier transform is

$$H(k) = \begin{cases} 0 & \text{if } |k| > k_c \\ 1 & \text{otherwise} \end{cases} \quad (2.18)$$

We shall refer to this as the SCK (Sharp cut-off in k-space) filter.

In inhomogeneous flows with turbulent rotational regions and irrotational regions, the two regions are separated by a sharp vorticity jump. In order to evaluate the ability of the SCK filter to smooth out jumps in the vorticity field, we apply it to a point vortex situated at the origin:

$$\omega(x,y) = \delta(x) \delta(y) \quad (2.19)$$

and

$$\bar{\omega}(x,y) = \frac{\sin[k_c x]}{\pi x} \frac{\sin[k_c y]}{\pi y} \quad (2.20)$$

$\bar{\omega}$ is plotted in Fig. 2.1.

First we note that this filter creates oscillations and negative vorticity, which are undesirable from a physical point of view. Second, those oscillations decay slowly (they go as x^{-1}), so the spreading into the irrotational region is excessive.

2.4.2 Gaussian (GS) Filter

Another filter that has been used by previous workers (Kwak et al., 1975) is the Gaussian spatial (GS) filter:

$$G(x-x') = \sqrt{\frac{\gamma}{\pi}} \frac{1}{\Delta} \exp\{-\gamma(x-x')^2/\Delta^2\} \quad (2.21)$$

where γ is a constant and Δ is the filter width.

Applying this filter to a point vortex situated at the origin, we get

$$\bar{\omega}(x,y) = \frac{\gamma}{\pi} \frac{1}{\Delta^2} \exp\left\{-\frac{\gamma}{\Delta^2} (x^2 + y^2)\right\} \quad (2.22)$$

$\bar{\omega}$ is plotted in Fig. 2.2.

We note that in this case we have created neither oscillations nor negative vorticity. By filtering the point vortex (Eqn. (2.19)), we have created another vortex with a Gaussian core of width Δ .

We conclude that a Gaussian filter smoothes out jumps better than the sharp cut-off filter. Therefore, the GS filter was used in the cases investigated in this work.

2.5 Computing Velocity Field from the Vorticity Field

When the vorticity equation is used, the velocity becomes a diagnostic variable; i.e., the time variation of the velocity is not given explicitly by the equations but can be deduced once the vorticity is known. To do so, we shall define a vector potential ψ_k (see Lamb, 1932) such that:

$$\bar{u}_i = \epsilon_{ijk} \frac{\partial}{\partial x_j} \psi_k \quad (2.23)$$

ψ_k can be chosen to be solenoidal; i.e.,

$$\frac{\partial}{\partial x_k} \psi_k = 0 \quad (2.24)$$

Taking the curl of (2.23) and using (2.24), we get

$$\frac{\partial^2}{\partial x_j \partial x_j} \psi_i = -\bar{\omega}_i \quad (2.25)$$

Solving the Poisson equation (2.25) and using (2.23), we get the velocity field from the vorticity field.

Note that the velocity field could have been deduced in another fashion by setting

$$\bar{\omega}_i = \epsilon_{ijk} \frac{\partial}{\partial x_j} \bar{u}_k \quad (2.26)$$

then taking the curl $\epsilon_{pqi} \partial/\partial x_q$ of Eqn. (2.26) to get:

$$\frac{\partial^2}{\partial x_j \partial x_j} \bar{u}_i = - \epsilon_{ijk} \frac{\partial}{\partial x_j} \bar{\omega}_k \quad (2.27)$$

and finally, solving the Poisson equation (2.27), we get the velocity field. This approach involves differentiation of the vorticity field, followed by a double integration, whereas the first approach of (2.25) involves double integration followed by differentiation (2.23). Numerically, the first approach is usually more desirable; but in our case the two approaches are equivalent. Eqns. (2.23)-(2.25) will be used in this study.

2.6 Summary

Neglecting the molecular viscosity, the filtered dynamical equations in vorticity form become

$$\frac{\partial \bar{\omega}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_j \bar{\omega}_i - \bar{u}_i \bar{\omega}_j) = - \frac{\partial}{\partial x_j} W_{ij} \quad (2.28)$$

$$\frac{\partial^2 \psi}{\partial x_j \partial x_j} = - \bar{\omega}_i \quad (2.25)$$

$$\bar{u}_i = \epsilon_{ijk} \frac{\partial}{\partial x_j} \psi_k \quad (2.23)$$

where W_{ij} is modeled as

$$W_{ij} = - \epsilon_{ijk} \frac{\partial}{\partial x_l} (2\nu_T \bar{S}_{kl}) \quad (2.15)$$

or

or

$$w_{ij} = - \frac{\partial}{\partial x_j} (v_T \bar{\omega}_i) + \frac{\partial}{\partial x_i} (v_T \bar{\omega}_j) \quad (2.16)$$

where

$$v_T = (c_v \Delta)^2 (\bar{\omega}_i \bar{\omega}_i)^{1/2} \quad (2.14)$$

and

$$\bar{s}_{ij} = \frac{1}{2} \left(\frac{\partial}{\partial x_j} \bar{u}_i + \frac{\partial}{\partial x_i} \bar{u}_j \right) \quad (2.12)$$

with

$$G(\underline{x}-\underline{x}') = \sqrt{\left(\frac{\gamma}{\pi}\right)^3} \frac{1}{\Delta_1 \Delta_2 \Delta_3} \exp \left\{ -\gamma \left[\frac{(x_1-x'_1)^2}{\Delta_1^2} + \frac{(x_2-x'_2)^2}{\Delta_2^2} + \frac{(x_3-x'_3)^2}{\Delta_3^2} \right] \right\} \quad (2.29)$$

and

$$\Delta = (\Delta_1 \Delta_2 \Delta_3)^{1/3}$$

It is in this form that the problem will be solved numerically.

Chapter 3

NUMERICAL METHOD

Analytical solutions of the governing equations discussed in the previous chapter can be found for only very special cases, none representing turbulence. Therefore, we propose using large computing machines to solve these equations for particular cases of interest. Numerical approximations of the governing equations require special care. In this chapter we discuss these approximations and present the methods we use to solve the difference approximation to the governing differential equations.

3.1 Notations

A region of continuous space is divided into a uniform rectangular mesh; h_i ($i=1,2,3$) represents the mesh width in the i^{th} direction. The mesh width need not be the same as the averaging width introduced in the previous chapters; we have used $\Delta_i = 2h_i$ and $\gamma = 6$. For details on the effects of the filter width on the computational results see Kwak et al. (1975).

We then write the ℓ -component of the filtered flow quantity f_ℓ at the n^{th} time step as

$$f_\ell^{(n)}(i,j,k), \quad \ell = 1,2,3 \quad (3.1)$$

where (i,j,k) are the mesh point index for (x,y,z) .

We define the operator notation $\delta/\delta\xi$ to be the numerical approximation to the continuous derivatives $(\partial/\partial\xi)$.

3.2 Numerical Approximation

Once space is discretized into mesh points, it remains to approximate the partial derivatives in terms of the values of the functions at those points. We have used two different approximation schemes: a fourth-order scheme and a pseudo-spectral method.

3.2.1 Fourth-Order Scheme

Using Taylor series expansions one can easily show that the approximation to the partial derivatives,

$$\frac{\delta \bar{u}}{\delta x_1} = \frac{1}{12h_1} \{ \bar{u}(i-2) - 8\bar{u}(i-1) + 8\bar{u}(i+1) - \bar{u}(i+2) \} \quad (3.2)$$

is fourth-order accurate, i.e., the error in this approximation is of $O(h^4)$. (For simplicity, the arguments j and k are not shown.)

If periodic boundary conditions are to be used, $\bar{u}$ can be represented by a discrete Fourier expansion (see next section).

$$\bar{u} = \sum_{\underline{n}} \hat{u}(\underline{k}) e^{i\underline{k} \cdot \underline{x}} \quad (3.3)$$

where, for $i = 1, 2, 3$,

$$k_i = \frac{2\pi}{N_i h_i} n_i = \text{wave number in the } x_i \text{ direction}$$

$$n_i = -\frac{N_i}{2}, \dots, 0, \dots, \frac{N_i}{2} - 1$$

$$N_i = \text{number of mesh points in } x_i \text{ direction}$$

$\hat{u}(\underline{k})$ is the discrete Fourier transform of $\bar{u}$. Taking the discrete Fourier transform of (3.2), we get

$$\begin{aligned} \frac{\delta \hat{u}}{\delta x_1} &= \frac{1}{12h_1} \left\{ e^{-i2h_1 k_1} - 8e^{-ih_1 k_1} + 8e^{ih_1 k_1} - e^{i2h_1 k_1} \right\} \hat{u} \\ &= \frac{i}{6h_1} \{ 8 \sin(h_1 k_1) - \sin(2h_1 k_1) \} \hat{u} \\ &= ik'_1 \hat{u} \end{aligned}$$

where

$$k'_1 = \frac{1}{6h_1} \{ 8 \sin(h_1 k_1) - \sin(2h_1 k_1) \} \quad (3.4)$$

is called the modified wave number.

Representation (3.4) allows us to evaluate the numerical approximation (3.2) for the range of wave numbers up to π/h_1 , the highest wave number that can be represented on a grid of size h_1 . The Fourier transform of the exact derivative is $ik_1 \hat{u}$, so that, by comparing the modified wave number k'_1 with k_1 , we see how well the approximation works (see Fig. 3.1).

A more important consequence of representation (3.4) is that it allows us to integrate numerically in a manner consistent with our difference approximation. In order to make this point clear, suppose we know the value, f , of the numerical approximation of the differential equation

$$\frac{\delta \bar{u}}{\delta x_1} = f \quad (3.5)$$

and we would like to find $\bar{u}$, which when fourth-order finite differenced, gives us f exactly (to machine round-off). One way to do this is to write

$$A\bar{u}(i) = f(i) \quad (3.6)$$

where

$$A = \frac{1}{12h_1} \begin{bmatrix} 0 & 8 & -1 & 0 & & & 0 & 1 & -8 \\ -8 & 0 & 8 & -1 & & & 0 & 0 & 1 \\ 1 & -8 & 0 & 8 & -1 & & & & \\ . & & & & & & & & \\ . & & & & & & & & \\ . & & & & & & & & \\ -1 & 0 & 0 & & & & & & \\ 8 & -1 & 0 & . & . & . & 1 & -8 & 0 \end{bmatrix}$$

for the case of periodic boundary conditions. This system of equations can then be solved in some standard way.

Another way to handle this problem is by taking the discrete Fourier transform of (3.5) to get

$$ik'_1 \hat{\bar{u}} = \hat{f} \quad (3.7)$$

Then, by solving for $\hat{u}$,

$$\hat{u} = \frac{\hat{f}}{ik_1} \quad (3.8)$$

multiplying (3.8) by $e^{ik_1 x_1}$, and summing over all k_1 , we obtain $\bar{u}$. In this case only the one-dimensional transform is needed. This method, which is much more powerful than the previous one when integration in more than one direction is needed, will be used extensively for the solution of the Poisson equations (2.21).

3.2.2 Pseudo-Spectral Method

Periodic boundary conditions

Suppose $f(x_1)$ is periodic in the x_1 direction with period L (in the following we shall consider the one-dimensional case) and satisfies the "Dirichlet condition", i.e.,

- $f(x_1)$ is defined at every point of the interval $0 \leq x_1 \leq L$,
- $f(x_1)$ is everywhere single-valued, finite, and sectionally continuous,
- $f(x_1)$ is of "bounded variation", i.e., $f(x_1)$ does not have an infinite number of maxima and minima.

It can be shown (Lanczos, 1956) that a function of this type can be expanded in a convergent Fourier series.

$$f(x_1) = \sum_{n_1=-\infty}^{\infty} \hat{f}(k_1) e^{ik_1 x_1} \quad (3.9)$$

where

$$k_1 = \frac{2\pi}{L} n_1 \quad n_1 = -\infty, \dots, 0, 1, \dots, \infty$$

and

$$\hat{f}(k_1) = \frac{1}{L} \int_0^L f(x_1) e^{-ik_1 x_1} dx_1 \quad (3.10)$$

Since computers cannot handle infinite series, we have to truncate (3.9). This is justifiable if $f(k_1)$ falls off rapidly for large $|k_1|$; this is the case of interest, since we filter the flow variables. Also, as mentioned before, we need to discretize in space. If N_1 mesh points are used in the x_1 direction, the discrete analogs of Eqns. (3.9) and (3.10) become:

$$f(x_1) = \sum_{n_1=-N_1/2}^{N_1/2-1} \hat{f}(k_1) e^{ik_1 x_1} \quad (3.11)$$

where

$$\begin{aligned} k_1 &= \frac{2\pi}{N_1 h_1} n_1 & n_1 &= -\frac{N_1}{2}, \dots, 0, \dots, \frac{N_1}{2} - 1 \\ x_1 &= j h_1 & j &= 0, \dots, N_1 - 1 \\ h_1 &= L/N_1 \end{aligned}$$

$$\hat{f}(k_1) = \frac{1}{N_1} \sum_{j=0}^{N_1-1} f(x_1) e^{-ik_1 x_1} \quad (3.12)$$

Fast algorithms (for $N_1 = 2^n$; $n = 1, 2, \dots$) have been developed (Fast Fourier Transform -- FFT) by various workers (Cooley and Tukey, 1965; Singleton, 1967) to evaluate the series (3.11) and (3.12) for the inverse-transform and the forward-transform, respectively. These will not be described in this work (we used a routine developed by Singleton, 1967).

If we regard the expansion (3.11) as an interpolating formula, so that we treat x_1 as a continuous variable, and differentiate the entire equation, we obtain

$$\frac{\delta f}{\delta x_1} = \sum_{n_1} \hat{f}(k_1) ik_1 e^{ik_1 x_1} \quad (3.13)$$

The expansion (3.13) can be considered an approximation to the partial derivatives. Thus, to compute the partial derivatives of $\bar{u}$, for the case of periodic boundary conditions, we proceed as follows: we find the discrete Fourier transform of the function in the direction in which the partial derivative is needed, i.e., we compute $\hat{f}(k_1)$ from $f(x_1)$.

Multiplying $f(k_1)$ by $ik_1 e^{ik_1 x_1}$ and summing over all k_1 , we obtain $\delta f / \delta x_1$. This is called the "pseudo-spectral" approach. This method has been analyzed by Lanczos (1956) and, with the development of techniques to compute the summations (3.11) rapidly, it has been proposed by Kreiss and Olinger (1973) as an approximation method and advocated by Orszag (1973) and Fox and Orszag (1973).

For the range of wave numbers that can be captured with a given spacing and number of grid points and for periodic boundary conditions, the pseudo-spectral method yields extremely accurate values of the partial derivatives (see Fig. 3.1).

The above method is limited to the case of periodic boundary conditions. However, the idea can be applied to other types of boundary conditions by using a set of orthogonal functions appropriate to the given boundary conditions.

$f = 0$ boundary conditions

If $f(x_1)$ is required to vanish at the boundary, i.e., $f(x_1) = 0$ for $x_1 = 0$ and $x_1 = L$, and is twice differentiable (a physically reasonable assumption), the Hilbert-Schmidt theory shows that its Fourier sine series

$$f(x_1) = \sum_{n_1=0}^{\infty} \hat{f}_{n_1}^s \sin \left[\frac{n_1 \pi}{L} x_1 \right] \quad (3.14)$$

where

$$\hat{f}_{n_1}^s = \frac{2}{L} \int_0^L f(x_1) \sin \left[\frac{n_1 \pi}{L} x_1 \right] dx_1 \quad (3.15)$$

is absolutely and uniformly convergent. As in the previous section, we shall use the discrete analogs to (3.14) and (3.15), i.e.,

$$f(x_1) = \sum_{n_1=0}^{N_1-1} \hat{f}_{n_1}^s \sin \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.16)$$

$$\hat{f}_{n_1}^s = \frac{2}{(N_1-1)h_1} \sum_{j=0}^{N_1-1} f(x_1) \sin \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.17)$$

where

$$n_1 = 0, \dots, N_1 - 1$$

$$h_1 = L/(N_1 - 1)$$

$$x_1 = jh_1, \quad j = 0, \dots, N_1 - 1$$

and $\hat{f}^s(n_1)$ is the Fourier sine transform of $f(x_1)$. By using the FFT routine, a technique to compute the summation in Eqns. (3.16) and (3.17) can be rapidly developed. A detailed development of the Fast Discrete Sine Transform (FDST) is given in Appendix B. Generally, the FDST requires twice as much computation (for a given number of mesh points) as does the FFT.

If we regard the expansion (3.16) as an interpolating formula, treating x_1 as a continuous variable, and differentiate, one obtains:

$$\frac{\delta f}{\delta x_1} = \sum_{n_1=0}^{N_1-1} \hat{f}^s(n_1) k_1 \cos \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.18)$$

where $k_1 = n_1 \pi / ((N_1-1)h_1)$. In order to be able to use (3.18) as an approximation formula for the partial derivatives, we need an FDST to find $\hat{f}^s(n_1)$; we also need a Fast Discrete Cosine Transform (FDCT). The discrete Fourier cosine series is defined in analogy to (3.16):

$$f(x_1) = \sum_{n_1=0}^{N_1-1} \hat{f}^c(n_1) \cos \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.19)$$

and

$$\hat{f}^c(n_1) = \frac{2}{(N_1-1)} \sum_{j=0}^{N_1-1} f'(x_1) \cos \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.20)$$

with

$$\hat{f}^c(n_1) = \begin{cases} \frac{1}{2} \hat{f}^c(n_1) & n_1 = 0, N_1 - 1 \\ \hat{f}^c(n_1) & n_1 \neq 0, N_1 - 1 \end{cases}$$

$$f'(x_1) = \begin{cases} \frac{1}{2} f(x_1) & x_1 = 0, L \\ f(x_1) & x_1 \neq 0, L \end{cases}$$

where $\hat{f}_1^c(n_1)$ is the Fourier cosine transform of $f(x_1)$.

Note that in (3.18) $\hat{f}^s(0) = \hat{f}^s(N_1 - 1) = 0$, making (3.18) exactly a discrete cosine transform of $k_1 \hat{f}^s(n_1)$.

By using the FFT routine, a technique of computing the summations in Eqn. (3.19) and (3.20) can be rapidly developed. A detailed development of the Fast Discrete Cosine Transform (FDCT) is given in Appendix B.

Thus, to compute the partial derivatives of a function which is zero at the boundary, we find its discrete sine transform $\hat{f}^s(n_1)$, multiply it by $n_1 \pi / [(N_1 - 1)h_1]$ and inverse transform using an FDCT routine. This method yields an extremely accurate approximation of the partial derivative when the function vanishes at the boundary, but its use is restricted to cases with a uniform mesh.

$\partial f / \partial x = 0$ boundary conditions

If $f(x_1)$ is our function whose partial derivative $\partial f / \partial x_1$ vanishes at the boundary, i.e., $\partial f / \partial x_1 = 0$ for $x_1 = 0$ and $x_1 = L$, then, by using arguments similar to those used before, it can be shown that its Fourier cosine series,

$$f(x_1) = \sum_{n_1=0}^{\infty} \hat{f}_1^c(n_1) \cos \left[\frac{n_1 \pi}{L} x_1 \right] \quad (3.21)$$

where

$$\begin{aligned} \hat{f}_1^c(n_1) &= \frac{2}{L} \int_0^L f(x_1) \cos \left[\frac{n_1 \pi}{L} x_1 \right] dx_1 \\ \hat{f}_1^c(n_1) &= \begin{cases} \hat{f}_1^c(n_1), & n_1 \neq 0 \\ \frac{1}{2} \hat{f}_1^c(n_1), & n_1 = 0 \end{cases} \end{aligned} \quad (3.22)$$

is uniformly and absolutely convergent.

Equations (3.19) and (3.20) are the discrete equivalents of the above equations. If we regard expansion (3.19) as an interpolating formula treating x_1 as a continuous variable, then differentiate, we obtain:

$$\frac{\delta f}{\delta x_1} = \sum_{n_1=0}^{N_1-1} -\hat{f}_1^c(n_1) k_1 \sin \left[\frac{n_1 \pi}{(N_1-1)h_1} x_1 \right] \quad (3.23)$$

Obviously (3.23) satisfies the conditions $\partial f / \partial x_1 = 0$ at $x_1 = 0$ and $x_1 = L$, and (3.23) is the discrete sine expansion of the partial derivative.

Thus, to compute the partial derivative of a function for which $\partial f / \partial x_1 = 0$ at the boundary, we find its discrete cosine transform $\hat{f}(n_1)$, multiply it by $-k_1$, and take the inverse transform using an FDST routine.

The three methods described in this section will be used extensively as our approximation tools.

3.3 Time Differencing

To advance in time, a second-order Adams-Bashforth method was used. This method has been used by previous workers (Kwak et al., 1975; Shaanan et al., 1975), and use of a higher-order method was not felt necessary.

If $\partial \bar{\omega}_i / \partial t = M_i$, the Adams-Bashforth formula for $\bar{\omega}_i$ at time-step $n + 1$ is

$$\bar{\omega}_i^{n+1} = \bar{\omega}_i^n + \Delta t \left(\frac{3}{2} M_i^{(n)} - \frac{1}{2} M_i^{(n-1)} \right) \quad (3.24)$$

In our case,

$$M_i = - \frac{\partial}{\partial x_j} \overline{(u_j \bar{\omega}_i - \bar{u}_i \bar{\omega}_j)} - \frac{\partial}{\partial x_j} \bar{w}_{ij}$$

Note that this is a two-step explicit method. It is started with the Euler method:

$$\bar{\omega}_i^1 = \bar{\omega}_i^0 + \Delta t M_i^{(0)} \quad (3.25)$$

3.4 Conservation Properties

As was pointed out by Phillips (1959), numerical integration of the finite-difference analog of the Navier-Stokes equations may introduce non-linear instabilities if proper care is not taken. Arakawa (1966), working with the two-dimensional vorticity equation, showed that by properly conserving vorticity, energy, and enstrophy ($\omega_i \omega_i$), these instabilities disappear. Lilly (1965), working with the primitive variables, developed a spatial-differencing scheme that conserves momentum and energy. By

conservation we mean that, in the absence of external forces and viscous dissipation, the only way that the momentum and kinetic energy in a control volume can change is by flow through the surface. This property must be retained by the numerical approximation. In the simple case of periodic boundary conditions, we have

$$\frac{d}{dt} \int_D \bar{u}_i dv = 0 \quad (\text{i.e., momentum conservation}) \quad (3.26)$$

$$\frac{d}{dt} \int_D \frac{1}{2} \bar{u}_i \bar{u}_i dv = 0 \quad (\text{i.e., energy conservation}) \quad (3.27)$$

It is usually easy to devise a numerical approximation to the dynamical equations in primitive form that conserves momentum, i.e., summation over the flow volume of the approximate equations would give the discrete equivalent of Eqn. (3.26). However, the difficulties arise when trying to show energy conservation, since in general the identity

$$u_i \frac{\partial}{\partial x_j} u_i = \frac{\partial}{\partial x_j} \frac{1}{2} u_i u_i \quad (3.28)$$

does not hold in finite-difference form.

Writing the equations of motion in the following form (Tennekes and Lumley, 1972):

$$\frac{\partial}{\partial t} u_i + u_j \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) = - \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) \quad (3.29)$$

$$\frac{\partial}{\partial x_i} u_i = 0 \quad (3.30)$$

and integrating over the flow volume, we get

$$\frac{\partial}{\partial t} \int_D u_i dv + \int_D u_j \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) dv = - \int_D \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) dv \quad (3.31)$$

For periodic boundary conditions, integration by parts yields:

$$\int_D u_j \frac{\partial}{\partial x_j} u_i dv = - \int_D u_i \frac{\partial}{\partial x_j} u_j dv = 0 \quad (\text{using (3.30)})$$

$$\int_D u_j \frac{\partial}{\partial x_i} u_j dv = - \int_D u_j \frac{\partial}{\partial x_i} u_j dv = 0$$

and

$$\oint \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) dv = 0$$

Hence Eqn. (3.31) reduces to Eqn. (3.26) and we have momentum conservation.

Now, multiplying Eqn. (3.29) by u_i , we get:

$$\frac{\partial}{\partial t} \frac{1}{2} u_i u_i = - u_i \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) \quad (3.32)$$

where the convective terms sum to zero by symmetry.

Integrating (3.32) over the entire domain yields:

$$\frac{\partial}{\partial t} \oint \frac{1}{2} u_i u_i dv = - \oint u_i \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) dv \quad (3.33)$$

For periodic boundary conditions, integration by parts yields:

$$\begin{aligned} \oint u_i \frac{\partial}{\partial x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) dv &= - \int \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) \frac{\partial}{\partial x_i} u_i dv \\ &= 0 \quad (\text{using (3.30)}) \end{aligned}$$

Hence Eqn. (3.33) reduces to Eqn. (3.27) and we have energy conservation.

We notice that, with the equations written in the form Eqn. (3.29), we did not need the identity (3.28) to show energy conservation from the dynamical equations in primitive form. The conservation properties were obtained by making use of only integration by parts and the continuity equation.

Consider the numerical approximation of Eqns. (3.29) and (3.30):

$$\frac{\partial}{\partial t} u_i + u_j \left(\frac{\delta u_i}{\delta x_j} - \frac{\delta u_j}{\delta x_i} \right) = - \frac{\delta}{\delta x_i} \left(\frac{p}{\rho} + \frac{1}{2} u_j u_j \right) \quad (3.34)$$

$$\frac{\delta}{\delta x_i} u_i = 0 \quad (3.35)$$

where we are using $\delta/\delta x_i$ to denote the numerical approximations to the partial derivatives $\partial/\partial x_i$, and the same approximations are used in both equations (3.34, 3.35) for any given independent variable. In order to

have long-term integration stability, Eqn. (3.34) should numerically conserve momentum and energy.

If we follow the steps used in deriving the conservation properties from Eqns. (3.29) and (3.30), we realize that the conservation properties will follow if we can establish numerical summation by parts. Consider the one-dimensional case, where we have, for periodic boundary conditions,

$$\int u(x) \frac{\partial}{\partial x} f(x) dx = - \int f(x) \frac{\partial}{\partial x} u(x) dx$$

The numerical analog of the above equation is:

$$\sum_{j=0}^{N-1} u(j) \frac{\delta}{\delta x} f(j) = - \sum_{j=0}^{N-1} f(j) \frac{\delta}{\delta x} u(j) \quad (3.36)$$

Expanding $u(j)$ in Fourier series, we get:

$$u(j) = \sum_{n=-N/2}^{N/2-1} \hat{u}(n) \exp(2\pi i j n / N) ; \quad j = 0, 1, \dots, N-1$$

where the $\hat{u}(n)$ are given by the inverse transform:

$$\hat{u}(n) = \frac{1}{N} \sum_{j=0}^{N-1} u(j) \exp(-2\pi i j n / N) ; \quad n = -\frac{N}{2}, \dots, \frac{N}{2}-1$$

Also,

$$\frac{\delta}{\delta x} f(j) = \sum_{n=-N/2}^{N/2-1} i k'(n) \left\{ \frac{1}{N} \sum_{j'=0}^{N-1} f(j') \exp(-2\pi i j' n / N) \right\} \exp(2\pi i j n / N) \quad (3.37)$$

where $k'(n)$ is the modified wave number. The modified wave numbers for the numerical methods we are using are

- $ik' = ik$ for pseudo-spectral, (3.38)
- $ik' = i \frac{1}{6h} [8 \sin(kh) - \sin(2kh)]$ (fourth-order approximation).

Substituting Eqn. (3.37) into the left-hand side of Eqn. (3.36) yields

$$\sum_{j=0}^{N-1} u(j) \frac{\delta}{\delta x} f(j) = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{j'=0}^{N-1} \sum_{n=-N/2}^{N/2-1} ik'(n) u(j) f(j') \cdot \exp(-2\pi i j' n / N) \exp(2\pi i j n / N)$$

Now, changing the summation index in the last sum from n to $-n$, we see that this expression will agree with the right-hand side of Eqn. (3.36), provided that:

$$k'(n) = -k'(-n) \quad (3.39)$$

$$k'\left(-\frac{N}{2}\right) = 0 \quad (3.40)$$

Condition (3.39) is satisfied by all the methods under consideration, and $k'(-N/2) = 0$ is true for the finite-difference method. The pseudo-spectral method cannot differentiate between $f = \exp(ij\pi)$ and $f = \exp(-ij\pi)$, and, due to this confusion at $n = -N/2$, $k'(-N/2)$ is set equal to zero for the pseudo-spectral method. Hence, summation by parts is obtained when (3.39) and (3.40) hold. Summing Equation (3.34) over all mesh points, using the generalization of (3.36) to three dimensions and using Eqn. (3.35) yields the numerical equivalent of (3.26). Multiplying Eqn. (3.34) by u_i , the nonlinear term in the left-hand side of (3.34) will sum to zero by symmetry; then, using as before the three-dimensional generalization of (3.36) and (3.35), summing over all mesh points will yield the numerical equivalent of Eqn. (3.27).

3.5 Differenced Vorticity Equations

In order to insure that the numerical approximation to the vorticity equations are equivalent to the numerical primitive equations, we must take the numerical curl of Eqn. (3.34). Before doing so, we note that, numerically,

$$\nabla \cdot \nabla \times \underline{V} = \epsilon_{ijk} \frac{\delta}{\delta x_i} \frac{\delta}{\delta x_j} V_k \quad (3.41)$$

$$\nabla \times \nabla S = \epsilon_{ijk} \frac{\delta}{\delta x_j} \frac{\delta}{\delta x_k} S \quad (3.42)$$

where $\underline{V}$ and S are any vector or scale, respectively, the above expressions are identically zero, if for each direction the same approximation is used for all operators.

The numerical curl of (3.34) is

$$\frac{\partial}{\partial t} \omega_i + \frac{\delta}{\delta x_j} (u_j \omega_i - u_i \omega_j) = 0 \quad (3.43)$$

Equation (3.43) conserves vorticity, i.e., summing it over all space the total vorticity in any control volume (subject to periodic boundary conditions) does not change with time. Hence in the form (3.34), the primitive equations also conserve vorticity.

The numerical divergence of (3.43) is

$$\frac{\partial}{\partial t} \frac{\delta}{\delta x_i} \omega_i = 0$$

Therefore, an $\underline{\omega}$ field solenoidal at time t will remain solenoidal at time $t + \Delta t$.

3.6 Poisson Equation

Having the vorticity field ω_i at time step n , we have to find the velocity field in order to be able to advance in time. To do so, we shall define a vector potential (also called the vector stream function) ψ_k , such that

$$\bar{u}_i = \epsilon_{ijk} \frac{\delta}{\delta x_j} \psi_k \quad (3.44)$$

ψ_i can be chosen to be solenoidal; i.e.,

$$\frac{\delta}{\delta x_i} \psi_i = 0 \quad (3.45)$$

Taking the curl of Eqn. (3.44) and using Eqn. (3.45), we get

$$\frac{\delta}{\delta x_j} \frac{\delta}{\delta x_j} \psi_i = -\bar{\omega}_i \quad (3.46)$$

The Poisson equations (3.46) will be integrated using the approach introduced in Section 3.2. For the case of periodic boundary condition, the discrete Fourier transform of Eqn. (3.46) is

$$-k_j' k_j' \hat{\psi}_i = -\hat{\omega}_i \quad (3.47)$$

where k'_i is the modified wave vector introduced in Section 3.2. Solving for $\hat{\psi}_i$, we have

$$\hat{\psi}_i = \frac{\hat{\omega}_i}{k'_j k'_j} \quad (3.48)$$

and by inverting the transform we obtain the stream vector consistent with our numerical differencing. It satisfies two conditions. First, the velocity field obtained using (3.44) will be solenoidal. We have in Fourier space:

$$k'_i \hat{u}_i = \epsilon_{ijk} k'_i k'_j \hat{\psi}_k = 0 \quad (3.49)$$

Second, taking the curl of (3.44), we have in Fourier space:

$$\begin{aligned} \hat{\omega}_i &= \epsilon_{ijk} i k'_j \hat{u}_k = -\epsilon_{ijk} \epsilon_{kpq} k'_j k'_p \hat{\psi}_q \\ &= -k'_j k'_j \hat{\psi}_i + k'_i k'_j \hat{\psi}_j \end{aligned} \quad (3.50)$$

Since $k'_j \hat{\psi}_j = 0$, (3.50) is exactly the Poisson equation (3.47).

3.7 Numerical Filtering

Examination of Eqn. (3.24) reveals that the only numerical problem left is the numerical evaluation of the $\overline{u_j \omega_i} - \overline{u_i} \overline{\omega_j}$ term. Since $\overline{u_j \omega_i} - \overline{u_i} \overline{\omega_j}$ can be computed easily, the problem is that of numerical filtering.

Filtering is the evaluation of a convolution integral

$$\overline{u_j \omega_i} = \int_{-\infty}^{+\infty} \overline{u_j \omega_i} G(x-x') dx' \quad (3.51)$$

If this integral is evaluated using conventional integration routines, the computation cost is prohibitive. Previous workers (Leonard, 1973; Kwak et al., 1975; Shaanan et al., 1975) argued that the filtered terms $\overline{u_j}(x')$ and $\overline{\omega_i}(x')$ are smooth, and they expanded those terms in a Taylor series about x . Using a Gaussian for $G(x)$, they obtained:

$$\overline{u_j \omega_i} = \overline{u_j \omega_i} + \frac{\Delta^2}{4\gamma} \nabla^2 (\overline{u_j \omega_i}) + O(\Delta^4) \quad (3.52)$$

and the $O(\Delta^2)$ term was called the Leonard term. The above approximation will require the use of a fourth-order, finite-differencing method (Kwak et al, 1975) or a modified second-order method (Shaanan et al., 1975) that yields the Leonard term as its truncation error. However, when higher-order methods are used the expansion (3.52) needs to be extended to higher orders, and the computational expense becomes prohibitive. When periodic boundary conditions are used, we can take the Fourier transform of Eqn. (3.51) to get:

$$\overline{\overline{u_j \omega_i}}^{\wedge} = (\overline{\overline{u_j \omega_i}}^{\wedge}) \hat{G} \quad (3.53)$$

Thus, given $\overline{u_i}$ and $\overline{\omega_i}$, one can compute the term $(\overline{\overline{u_j \omega_i}}^{\wedge})$, multiply it by $\hat{G}$, then simply invert the transform to obtain $\overline{\overline{u_j \omega_i}}$.

When $\overline{u_j \omega_i}$ vanishes at the boundaries, i.e., $\overline{u_j \omega_i} = 0$ at $x = 0$ and $x = L$, we can expand it in a Fourier sine series. Taking the one-dimensional case, for simplicity, we set

$$\overline{u_j \omega_i} = \sum_{n=0}^{\infty} (\overline{\overline{u_j \omega_i}}^{\wedge s}) \sin\left(\frac{n\pi}{L} x\right) \quad (3.54)$$

Substituting (3.54) in (3.51), we get

$$\overline{\overline{u_j \omega_i}} = \int_{-\infty}^{+\infty} \sum_{n=0}^{\infty} (\overline{\overline{u_j \omega_i}}^{\wedge s}) \sin\left(\frac{n\pi}{L} (x-x')\right) G(x') dx'$$

Since the series (3.54) is absolutely and uniformly convergent, we can take the summation outside the integration to obtain

$$\begin{aligned} \overline{\overline{u_j \omega_i}} = \sum_{n=0}^{\infty} (\overline{\overline{u_j \omega_i}}^{\wedge s}) \left\{ \int_{-\infty}^{+\infty} \sin\left(\frac{n\pi}{L} x\right) \cos\left(\frac{n\pi}{L} x'\right) G(x') dx' \right. \\ \left. + \int_{-\infty}^{+\infty} \cos\left(\frac{n\pi}{L} x\right) \sin\left(\frac{n\pi}{L} x'\right) G(x') dx' \right\} \end{aligned}$$

If $G(x)$ is an even function, which is the case when (2.21) is used, the second term in the bracket vanishes and one obtains

$$\begin{aligned} \overline{\overline{u_j \omega_i}} &= \sum_{n=0}^{\infty} (\overline{\overline{u_j \omega_i}}^{\wedge s}) \left\{ \int_{-\infty}^{+\infty} G(x') \cos\left(\frac{n\pi}{L} x'\right) dx' \right\} \sin\left(\frac{n\pi}{L} x\right) \\ &= \sum_{n=0}^{\infty} (\overline{\overline{u_j \omega_i}}^{\wedge s}) \hat{G}^c \sin\left(\frac{n\pi}{L} x\right) \end{aligned} \quad (3.55)$$

where

$$\hat{G}^c = \int_{-\infty}^{+\infty} G(x') \cos\left(\frac{n\pi}{L} x'\right) dx'$$

is the Fourier cosine transform of the Gaussian filter.

What Eqn. (3.55) tells us is that, for the case in which $\overline{u_j \omega_i} = 0$ for $x = 0$ and $x = L$, $\overline{u_j \omega_i}$ can be computed by the following procedure: we first compute the Fourier sine transform $(\overline{u_j \omega_i})^{\wedge s}$ of $\overline{u_j \omega_i}$, and then multiply it by the Fourier cosine transform $\hat{G}^c$ of the filter, to obtain the Fourier sine transform $(\overline{u_j \omega_i})^{\wedge s}$ of $\overline{u_j \omega_i}$. Finally, inverting the sine transform, we obtain $\overline{u_j \omega_i}$.

Similarly, it can be shown that, for the case in which $\frac{\partial}{\partial x} \overline{u_j \omega_i} = 0$ at $x = 0$ and $x = L$, we have

$$\overline{u_j \omega_i} = \sum_{n=0}^{\infty} (\overline{u_j \omega_i})^{\wedge c} \hat{G}^c \cos\left(\frac{n\pi}{L} x\right) \quad (3.56)$$

or

$$\frac{\hat{c}}{(\overline{u_j \omega_i})} = (\overline{u_j \omega_i})^{\wedge c} \hat{G}^c$$

By the use of the FFT, FDST, and FDCT, "exact" filtering can be obtained for all boundary conditions of interest with acceptable computational speed.

An important property required of a filter is that the filtered value of a constant must be the same constant. Numerically, it is desirable to preserve this property, which is equivalent to requiring the integral of the filter function be unity or $\hat{G}(0) = 1$. The exact continuous Fourier transform of (2.21) is

$$\hat{G}(k) = \exp\left(-\frac{\Delta^2}{4\gamma} k^2\right) \quad (3.57)$$

When $\hat{G}(k)$ is discretized, we get

$$\hat{G}_D(k) = \exp\left(-\frac{\Delta^2}{4\gamma} \left(\frac{2\pi}{L} n\right)^2\right), \quad n = 0, \pm 1, \pm 2, \dots \quad (3.58)$$

Hence $\hat{G}_D(0) = 1$.

Another property required of a filter is that it smooth out jumps (see Section 2.4) without introducing oscillations. We have modeled the situation with a top-hat function:

$$f(x) = \begin{cases} 1 & x_1 \leq x \leq x_2 \\ 0 & \text{otherwise} \end{cases} \quad (3.59)$$

Analytically, we have

$$\bar{f}(x) = \frac{1}{2} (\text{erf}(x_1 - x) - \text{erf}(x_2 - x))$$

which is a smooth function with no oscillations.

When (3.59) is discretized and filtered numerically using $\hat{G}_D(k)$, the top-hat function, Eqn. (3.59), is smoothed out (see Fig. 3.2). However, small oscillations are introduced. This is due to the fact that the discrete inverse transform of (3.58) is not smooth. For this reason we have used a discrete Gaussian in x-space,

$$G_D(x) = \frac{1}{A} \exp\left(-\gamma \frac{(nh)^2}{\Delta^2}\right) \quad (3.60)$$

where

where

$$A = \sum_n \exp\left(-\gamma \frac{(nh)^2}{\Delta^2}\right) \quad (3.61)$$

as our filter function. The oscillations in the x-space (see Fig. 3.2) do not appear when this filter is used.

3.8 Summary

The dynamical equations in vorticity form will be solved as follows:

$$\omega_i^{n+1} = \omega_i^n + \Delta t \left(\frac{3}{2} M_i^n - \frac{1}{2} M_i^{n-1} \right) \quad (3.24)$$

where

$$M_i = -\frac{\delta}{\delta x_j} \overline{(\bar{u}_j \bar{\omega}_i - \bar{u}_i \bar{\omega}_j)} - \frac{\delta}{\delta x_j} W_{ij}$$

$$W_{ij} = -\epsilon_{ijk} \frac{\delta'}{\delta x_l} (2v_T \bar{s}_{kl}) \quad (3.25)$$

or

$$W_{ij} = -\frac{\delta'}{\delta x_j} (v_T \bar{\omega}_i) + \frac{\delta'}{\delta x_i} (v_T \bar{\omega}_j) \quad (3.26)$$

$$v_T = (c_v \Delta)^2 (\bar{\omega}_i \bar{\omega}_i)^{1/2} \quad (3.27)$$

and

$$\bar{s}_{ij} = \frac{1}{2} \left(\frac{\delta' \bar{u}_i}{\delta x_j} + \frac{\delta' \bar{u}_j}{\delta x_i} \right) \quad (3.28)$$

The numerical differencing $\frac{\delta'}{\delta x}$ used to compute the terms in the model (W_{ij}) need not be of the same order as the numerical differencing $\frac{\delta}{\delta x}$ used to compute the terms in the momentum equation. Filtering of the terms $\overline{u_j \omega_i - u_i \omega_j}$ is achieved using the method described in Section 3.7.

Chapter 4

DECAY OF ISOTROPIC TURBULENCE

4.1 Background

In order to assess the feasibility of using the vorticity equations as the governing equations for turbulent flows, we applied the computational methods described in Chapter 3 to the simplest problem in turbulence, namely, the decay of homogeneous isotropic turbulence. This flow was also used to determine the value of the subgrid scale model constant for use in subsequent calculations of other flows.

The grid turbulence experiment of Comte-Bellot and Corrsin (1971) was used as the "target" for our numerical predictions. When viewed in a coordinate frame moving with the mean velocity, this experiment approximates homogeneous isotropic turbulence.

This study was presented in an earlier report (P. Moin et al., 1978) and is rediscussed in this work to support the argument that model ω -2 used in conjunction with the vorticity equations produces similar results to those obtained using the more commonly used model ω -1. The contributions of Mr. P. Moin are gratefully acknowledged.

4.2 Initial Conditions

We started with an initial field that is divergence-free and has a spectrum obtained by filtering the experimental spectrum at the non-dimensional experimental time $T = U_0 t / M = 42$. $U_0 = 10$ cm/sec is the experimental free-stream air speed, $M = 5.08$ cm is the size of the experimental turbulence-generating grid, and t is the real time in seconds. The initial field was otherwise random. The generation of such a field is discussed in detail by Kwak et al. (1975) and will be briefly outlined herein. The filtered field is generated in k -space by setting:

$$\hat{u}_i(\underline{k}) = \left(\frac{2\pi}{L}\right)^3 \left(\frac{\overline{E(k)}}{2\pi k^2}\right)^{1/2} (aA_i + ibB_i) \quad (4.1)$$

where

$\bar{E}(k)$ = filtered experimental energy spectrum at time $T = U_0 t/M = 42$,

$$a = \cos(\theta)$$

$$b = \sin(\theta)$$

where θ is a random angle, A_i and B_i are unit vectors picked such that $A_i k'_i = B_i k'_i = 0$, otherwise random.

To insure that (4.1) is the Fourier transform of a real field, we must have

$$\hat{u}_i(k) = \hat{u}_i^*(-k) \quad (4.2)$$

where $*$ indicates complex conjugate. Now, by inverse transforming $\hat{u}_i$, $\bar{u}_i$ is obtained.

Using the above initial field, we shall use the methods of Chapters 2 and 3 to predict the spectrum at $T = 98$. The predicted spectrum will be compared with the filtered experimental spectrum at $T = 98$.

4.3 Selection of C_v

The model constant was obtained by matching the computational rate of filtered energy decay to that of the experiment (Fig. 4.1). The values of the constants obtained using different numerical schemes and different models were in most cases within ten percent of each other ($C_v = 0.2 \pm 0.02$, see Table 4.1).

4.4 Results

Under the assumption that the computational box size is large compared to the scale of the energy-containing motions, we can use periodic boundary conditions in all three directions. A uniform cubic mesh system was used with N , the mesh number in each direction, and h , the mesh spacing, chosen such that the computation captures as much of the turbulence energy as possible (Kwak et al., 1975). We used the sets

$$N = 16, \quad h = 1.5 \text{ cm}, \quad t = 6.25 \times 10^{-3} \text{ sec}$$

and

$$N = 32, \quad h = 1.0 \text{ cm}, \quad t = 6.25 \times 10^{-3} \text{ sec}$$

When periodic boundary conditions are used, it was shown in Chapter 3 that the pseudo-spectral method is more powerful than any finite-difference method. However, when the periodic pseudo-spectral methods cannot be used, we may have to use finite-difference methods. Since one of our objectives is to determine the model constant for the vorticity equations, both the fourth-order finite differencing and the pseudo-spectral methods were applied to the case of isotropic homogeneous turbulence.

4.4.1 Fourth-Order Finite Differences

Figure 4.2 shows the energy spectrum obtained by fourth-order finite-differencing the vorticity equation, using model $\omega-1$ (Eqn. (2.1)) for the subgrid-scale model, on a 16^3 mesh. Our results compare well with the experimental results up to wave number 2.5, after which the inaccuracy of fourth-order differencing begins to show. Fourth-order differencing the primitive equations (Kwak et al., 1975; Moin et al., 1978) produced good agreement with the filtered experimental results using the primitive variable version of this model. This shows that the vorticity approach is equivalent to the primitive variable method. Thus the use of the vorticity equations is definitely feasible in turbulent flow computations.

4.4.2 Pseudo-Spectral Method

Figures 4.3-4.6 show the energy spectra obtained using the pseudo-spectral method, with 16^3 mesh. Fig. 4.3 shows the results obtained using model $\omega-1$ (Eqn. (2.15)). We note that for $k \geq 1$ the computed results are considerably lower than the experimental values. This indicates that the subgrid-scale model is draining too much energy from the small structures, and, since our total energy is equal to that of the filtered experimental value, too little energy is taken out from the large structures. In this case, we used the pseudo-spectral approximation to calculate the subgrid scale terms as well as the other terms.

Figure 4.4 shows the energy spectrum obtained using second-order central differencing to approximate the derivatives appearing in the subgrid-scale model (see Section 3.8):

$$\frac{\delta' f}{\delta x} = \frac{f(i+1) - f(i-1)}{2h}$$

We note a considerable improvement in the spectrum, except for a small accumulation of energy at the extreme (high wave number) end of the spectrum, which was present to a lesser extent in Fig. 4.2.

Figures 4.5 and 4.6 are the results from a 16^3 computation using the pseudo-spectral method and model $\omega-2$ (Eqn. (2.16)) for the subgrid-scale model. We note the same behavior in Fig. 4.5 as in Fig. 4.3; the computed spectrum falls below the experimental spectrum, indicating that using the pseudo-spectral method to compute the spatial derivatives in the subgrid-scale model damps too much energy in the wave number range $k \geq 1$. Using second-order finite differencing to compute the partial derivatives in the model $\omega-2$ (Eqn. (2.16)), we obtain a significant improvement in the computed spectrum (Fig. 4.6). These results are similar to the results obtained using model $\omega-1$, indicating that the two models are equally good.

Figure 4.7 shows the energy spectrum obtained from a 32^3 pseudo-spectral calculation, using second-order finite differencing to compute the partial derivatives in model $\omega-2$. The results are similar to those of the 16^3 computation.

It can be concluded from these results that the vorticity equations provide a satisfactory basis for the simulation of homogeneous isotropic turbulence. Both models $\omega-1$ and $\omega-2$ produce similar results. Model $\omega-2$, given by Eqn. (2.16), will be used in the following computations, due to the computational advantages it offers over model $\omega-1$ (see Section 2.3). Finally, a relatively coarse 16^3 mesh is sufficient to capture interesting features of the homogeneous isotropic turbulence, and no significant improvement in the energy spectrum was obtained by using a 32^3 mesh system.

4.5 Computational Details

The calculations described above were executed on the CDC-7600 at NASA-Ames Research Center, using programs written in Fortran. The total storage requirements (octal) were as follows:

16³ Calculation

Large Core Memory:	Fourth-order	310,360
	Pseudo-spectral	230,000
Small Core Memory:	Fourth-order	104,465
	Pseudo-spectral	61,334

32³ Calculation

Large Core Memory:	Pseudo-spectral	1,110,000
Small Core Memory:		126,605

The computing time per computational time step was approximately as follows:

16³ Calculation

Fourth order	=	2.5 sec CPU time
Pseudo-spectral	=	4.0 sec CPU time.

32³ Calculation

Pseudo-spectral	=	34 sec CPU time.
-----------------	---	------------------

Chapter 5

MIXING LAYER: TWO-DIMENSIONAL COMPUTATION

5.1 Preview

It is well documented (Winant and Browand (W&B), 1974; Brown and Roshko (B&R), 1974; Konrad, 1976; Dimotakis and Brown (D&B), 1976) that in some cases the spatially developing mixing layer contains coherent structures (in the terminology of B&R) or discrete vortices (in the terminology of W&B). In these experiments, the mixing layer grows via the interaction of neighboring vortex-like structures that rotate around and combine with each other to form a similar but larger structure (see Fig. 5.1). This mechanism is called vortex pairing. In this chapter we study the vortex-pairing mechanism by perturbing an infinite array of vortices. The effect of the initial perturbation on the roll-up is discussed. All cases treated in this chapter are completely two-dimensional; three-dimensional cases are discussed in the next chapter.

5.2 Some Experimental Results

The mixing layer is generated in a laboratory by bringing together two streams of fluid of different streamwise velocity (see Fig. 5.2). The measured mean velocity profiles, at different streamwise positions, are self-similar and can be fitted by an error-function (Spencer and Jones (S&J), 1971):

$$\frac{u}{u_1} = r + \frac{(1-r)}{2} \left(1 - \operatorname{erf}(\sigma(\eta - \eta_0)) \right) \quad (5.1)$$

where

- $r = u_2/u_1,$
- $u_1 =$ velocity of the high-speed side,
- $u_2 =$ velocity of the low-speed side,
- $\eta = z/(x - x_0)$
- $\sigma =$ spread parameter
- $z =$ cross-flow coordinate,

x = streamwise coordinate, and
 x_0 = virtual origin of the layer.

Rearranging (3.1) and normalizing the velocity on $\Delta u = u_1 - u_2$, we get

$$\frac{u - U}{\Delta u} = 0.5 \operatorname{erf}(\sigma(\eta - \eta_0)) \quad (5.2)$$

where $U = (u_1 + u_2)/2$ is the mean velocity. The spread parameter σ is a function of r , and the spread data can be fitted by the expression:

$$\frac{\sigma}{\sigma_0} = \frac{1+r}{1-r} \quad (5.3)$$

where σ_0 is the spread parameter for $r = 0$. S&J report $\sigma_0 = 11$ for other values of σ_0 ; see Table 1.1.

Defining the momentum thickness, θ , to be

$$\begin{aligned} \theta &= \frac{1}{(\Delta u)^2} \int_{-\infty}^{+\infty} (u - u_2)(u_1 - u) dz \\ &= \int_{-\infty}^{+\infty} \left(\frac{1}{4} - \frac{(u - U)^2}{(\Delta u)^2} \right) dz \end{aligned} \quad (5.4)$$

and substituting (5.2) in (5.4), we get

$$\theta = \frac{x - x_0}{\sigma \sqrt{2\pi}} = \frac{x - x_0}{\sigma_0} \frac{\Delta u}{2 \sqrt{2\pi} U} \quad (5.5)$$

$$\sigma = \frac{x - x_0}{\theta \sqrt{2\pi}} \quad (5.6)$$

Since σ is constant, Eqn. (5.5) shows that the momentum thickness grows linearly with x .

Substituting (5.6) in (5.2), we get:

$$\frac{u - U}{\Delta u} = 0.5 \operatorname{erf}\left(\frac{(z - z_0)}{\theta \sqrt{2\pi}}\right) \quad (5.7)$$

Due to computer limitations, one cannot set up a uniform grid that covers the length of the experimental set-up (1.8 m for the W&B case) and

at the same time resolves the large-eddy scale ($\sim 1-4$ cm). We propose to use a uniform grid that moves with the mean speed U . The size of the computational domain is chosen so that two vortices are captured in the initial field; i.e., we can imagine that we are following the fluid in the dashed box in Fig. 5.1 as it moves downstream.

In our frame the layer will develop in time rather than space. We shall in fact be studying a portion of a time-developing mixing layer. This layer can be thought of as being created by having two infinite counter-moving streams of velocity $\pm \Delta u/2$ brought in contact suddenly at $T = 0$. For this flow, the mean quantities will be horizontal planar-averaged quantities; for example, the mean velocity profile will be defined as

$$\langle \bar{u} \rangle_{xy} = \frac{1}{A} \int_A \bar{u}(x, y, z, t) dx dy \quad (5.8)$$

The momentum thickness, defined as

$$\theta(t) = \int \left[\frac{1}{4} - \left(\frac{\langle \bar{u} \rangle_{xy}}{\Delta u} \right)^2 \right] dz \quad (5.9)$$

will be a function of time instead of space. According to the Taylor hypothesis, the state of the flow at the experimental streamwise distance x is the same as that of the computed layer at the computational time variable t . The variables x and t are related by the expression:

$$x = Ut \quad (5.10)$$

Substituting (5.10) in (5.5), we get an expression for the expected momentum thickness of the time-developing layer:

$$\theta(t) = \frac{t - t_0}{\sigma_0^2 \sqrt{2\pi}} \Delta u \quad (5.11)$$

Equation (5.11) shows that $\theta(t)$ should grow linearly with time, with

$$\frac{d\theta}{\Delta u dt} = \frac{1}{\sigma_0^2 \sqrt{2\pi}} \quad (5.12)$$

5.3 Boundary Conditions

The coordinate system used is shown in Fig. 5.3, where the x-direction is the streamwise direction, the y-direction is the spanwise direction, and the z-direction is the cross-flow direction. We shall use periodic boundary conditions in the x- and y-directions; this is allowed if the size of the computational box is sufficiently greater than the integral scale in a given direction. At a large enough z location the flow is essentially horizontal and uniform. We can use no stress boundary conditions in the z-direction (i.e., $\partial u/\partial z = \partial v/\partial z = w = 0$ at $z = 0$ and $z = L$) if the boundaries of our box in this direction are sufficiently far from the center of the layer. This will allow us to expand the velocity fields as follows:

$$\bar{u} = \sum_n \sum_{k_2} \sum_{k_1} \hat{u}(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \cos\left(\frac{n\pi z}{L_3}\right) \quad (5.13)$$

$$\bar{v} = \sum_n \sum_{k_2} \sum_{k_1} \hat{v}(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \cos\left(\frac{n\pi z}{L_3}\right) \quad (5.14)$$

$$\bar{w} = \sum_n \sum_{k_2} \sum_{k_1} \hat{w}(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \sin\left(\frac{n\pi z}{L_3}\right) \quad (5.15)$$

and the vorticity fields as follows:

$$\bar{\omega}_1 = \sum_n \sum_{k_2} \sum_{k_1} \hat{\omega}_1(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \sin\left(\frac{n\pi z}{L_3}\right) \quad (5.16)$$

$$\bar{\omega}_2 = \sum_n \sum_{k_2} \sum_{k_1} \hat{\omega}_2(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \sin\left(\frac{n\pi z}{L_3}\right) \quad (5.17)$$

$$\bar{\omega}_3 = \sum_n \sum_{k_2} \sum_{k_1} \hat{\omega}_3(k_1, k_2, n) e^{i(k_1 x + k_2 y)} \cos\left(\frac{n\pi z}{L_3}\right) \quad (5.18)$$

The pseudo-spectral method will be used to approximate the partial derivatives. The numerical technique was discussed in Chapter 3.

5.4 Initial Conditions

We want to prescribe an initial profile that corresponds to a pair of vortices. It has been shown in Chapter 2 that filtering a line vortex produces a vortex with a Gaussian distribution of vorticity in the core. We shall use this fact to generate our initial conditions.

The initial conditions are generated by starting with two line vortices in the spanwise direction at $(x = x_1, z = L_3/2)$ and $(x = x_2, z = L_3/2)$ (see Fig. 5.4), and filtering in the x - z plane with the relatively wide Gaussian filter:

$$G(x, z) = \frac{1}{A_1 A_3} \exp\left(-\frac{x^2}{6h_1^2} - \frac{z^2}{6h_3^2}\right) \quad (5.19)$$

where h_i is the mesh size in the i -th direction ($i = 1, 3$) and A_i ($i = 1, 2$) is defined by Eqn. (3.61). This will produce the vorticity field:

$$\bar{\omega}_2 = C_1 \frac{1}{A_1 A_3} \left\{ \exp\left(-\frac{(x-x_1)^2}{6h_1^2}\right) + \exp\left(-\frac{(x-x_2)^2}{6h_2^2}\right) \right\} \exp\left(-\frac{(z-L_3/2)^2}{6h_3^2}\right) \quad (5.20)$$

$$0 \leq x < L_1, \quad 0 \leq z < L_3$$

$$\bar{\omega}_1 = \bar{\omega}_3 = 0$$

$$\bar{\omega}_2(x_1, z) = \bar{\omega}_2(x + nL_1, z) \quad n = \pm 1, \pm 2, \dots (\text{periodicity})$$

where C_1 is an arbitrary constant that adjusts the strength of the vortices. Note that these vortices can be elliptical; they are h_1/h_3 times as long in the streamwise direction as in the cross flow direction.

Equations (5.20) correspond to a perturbed infinite array of vortices with a perturbation parameter β equal to:

$$\beta = \frac{1}{2} - \frac{|x_1 - x_2|}{L_1} \quad (5.21)$$

$\beta = 0$ corresponds to a uniform (unperturbed) vortex array, and we need to deal only with the case $\beta > 0$.

Figures 5.6a-f show constant vorticity contours for $\beta = 6/16, 5/16, 4/16, 3/16, 2/16$, and $1/16$. Note that for large β the vorticity contours look like those for a single distorted vortex.

5.5 Mesh Size Selection

We have shown in Chapter 4 that a $16 \times 16 \times 16$ mesh system can resolve isotropic homogeneous turbulence with acceptable accuracy. For the cases considered in this chapter there are no variations in the spanwise direction. We dropped the number of meshes in the spanwise direction to $N_2 = 4$, the minimum number of meshes that our three-dimensional code was designed to handle. In the cross-flow direction the mesh number was increased to $N_3 = 33$ in order to allow the layer to grow in this direction. This gives a total number of mesh points of $N_1 \times N_2 \times N_3 = 16 \times 4 \times 33 = 2112$.

The spanwise vorticity is defined by

$$\bar{\omega}_2 = \frac{\partial}{\partial z} \bar{u} - \frac{\partial}{\partial x} \bar{v} \quad (5.22)$$

Averaging (5.22) over x - y planes and using periodic boundary conditions, we get

$$\langle \bar{\omega}_2 \rangle_{xy} = \frac{d}{dz} \langle \bar{u} \rangle_{xy} \quad (5.23)$$

If we substitute in (5.23) the vorticity distribution given by Eqn. (5.20), we get:

$$\frac{d}{dz} \langle \bar{u} \rangle_{xy} = C_1 \frac{2}{L_1 A_3} \exp\left(-\frac{(z-L_3/2)^2}{6h_3^2}\right) \quad (5.24)$$

This ordinary differential equation can be solved together with the boundary condition:

$$\langle \bar{u} \rangle_{xy} = 0 \quad \text{at} \quad z = L_3/2 \quad (5.25)$$

The solution is obtained by simple integration:

$$\langle \bar{u} \rangle_{xy} = \frac{C_1}{L_1} \operatorname{erf}\left(\frac{z-L_3/2}{\sqrt{6} h_3}\right) \quad (5.26)$$

Non-dimensionalizing the velocity Δu , we get:

$$\langle \bar{u} \rangle_{xy} = 0.5 \operatorname{erf}\left(\frac{z-L_3/2}{\sqrt{6} h_3}\right) \quad (5.27)$$

Equating Eqns. (5.26) and (5.27) and solving for C_1 , we get:

$$C_1 = 0.5 L_1 \quad (5.28)$$

The length scales are non-dimensionalized on the momentum thickness. The mesh size was chosen such that the initial momentum thickness is equal to unity. Substituting (5.27) in Eqn. (5.9), we get:

$$\theta_{in} = \frac{\sqrt{6}}{\sqrt{2\pi}} h_3 = 1$$

and solving for h_3 , we obtain

$$h_3 = \sqrt{\frac{2\pi}{6}} = 1.023 \quad (5.29)$$

The mesh size in the streamwise direction was set equal to:

$$h_1 = \frac{4}{3} h_3 = 1.364 \quad (5.30)$$

The non-dimensional time step was picked up to be equal to:

$$\Delta T = \frac{\Delta u \Delta t}{\theta_{in}} = 0.0799 \quad (5.31)$$

which yields a Courant number such that:

$$N_c = U_\infty \frac{\Delta t}{h_1} \leq 0.03$$

which is well within the stability criterion and assures that the error caused by the time advancement will be acceptably small.

The mesh size in the spanwise direction is irrelevant for the cases considered in this chapter. We have set $h_2 = h_3$.

5.6 Selection of β

We have shown in Section 5.2 that, to accord with the experimental observations, the momentum thickness $\theta(t)$ must grow linearly with time; and, using $\sigma_o = 11$ (S&J), we expect:

$$\frac{d\theta}{\Delta u dt} = \frac{1}{\sigma_o^2 \sqrt{2\pi}} = 0.018 \quad (5.32)$$

We have run a series of calculations for different values of β . Fig. 5.5 shows the momentum thickness θ/θ_{in} plotted vs. T for the cases run. For the highly perturbed cases, $\beta \geq 4/16$, the momentum thickness $\theta(t)$ does not grow linearly in time. However, for $\beta = 3/16, 2/16$, and $1/16$, $\theta(t)$ does grow linearly in time, with $d\theta/\Delta u dt = 0.020, 0.015$, and 0.009 , respectively.

Figures 5.6 and 5.7 show constant vorticity (contour) plots for the various cases at times $T = 0$ and $T = 16.78$, respectively. Figs. 5.6a-c and 5.7a-c show that for large β we have essentially one elliptical vortex which grows "fatter" in time, to become more or less circular at time $T = 16.78$. Figs. 5.6d-f show that for small β , we have initially two distinct vortices; these vortices draw closer and rotate around each other (Figs. 5.7d-f). For the case $\beta = 3/16$, the two vortices merge to form one vortex at time $T = 16.78$ (Fig. 5.7d).

The above observations indicate that case $\beta = 3/16$ gives results comparable to the experimental observations. The spread parameter σ_o obtained for $\beta = 3/16$ is equal to

$$\sigma_o = \frac{1}{\frac{d\theta}{\Delta u dt} \sqrt{2\pi}} = 9.97$$

which is within 10% of the experimental results of S&J.

5.7 Mean Velocity Profiles

The mean velocity profile $\langle \bar{u} \rangle_{xy}$ defined by Eqn. (5.8) is a function of z and T . Fig. 5.8 shows $2\langle \bar{u} \rangle_{xy}/\Delta u$ plotted vs. z/θ at $\Delta T = 2.4$ intervals, for $\beta = 3/16$. The profiles collapse into one, indicating self-similarity of the mean velocity profiles.

Self-similarity is also observed in the experimental data. Thus, as far as the mean profile is concerned, the data can be fit by pairing vortices with $\beta = 3/16$.

5.8 Mean Turbulent Intensity Profiles

In our computational box, the non-dimensional mean turbulence intensity is defined as

$$\frac{q^2}{2(\Delta u)^2} = \frac{1}{2(\Delta u)^2} \langle (\bar{u} - \langle \bar{u} \rangle_{xy})^2 + (\bar{v} - \langle \bar{v} \rangle_{xy})^2 + (\bar{w} - \langle \bar{w} \rangle_{xy})^2 \rangle_{xy} \quad (5.33)$$

where $\langle \rangle_{xy}$ are planar averages defined by Eqn. (5.8).

Figure 5.9 shows the mean turbulence intensity plotted vs. z/θ , for the case $\beta = 3/16$, at $\Delta T = 2.4$ intervals. We note that the turbulence intensity decays slightly at the early stages of the pairing and then reaches a self-similar situation.

Compared with the experimental results, our peak intensity $q^2/2(\Delta u)^2|_{\max} = 2.06 \times 10^{-2}$ is substantially lower than the experimental value reported by S&J (3.5×10^{-2}). The low value of the maximum turbulence intensity is due to the fact that we did not take into account the subgrid scale contributions, and that our field is strictly two-dimensional, whereas in reality spanwise fluctuations are present in the experiment of S&J.

5.9 Summary

It is interesting to note that vortex pairing is capable of producing self-similar mean velocity and turbulence intensity profiles, and a linear growth of the momentum thickness that compare with experimental results (for $\beta = 3/16$). We note that, due to periodic boundary conditions, once the vortices have paired we get a uniform vortex array ($\beta = 0$) and the pairing and layer growth stop. If we want the pairing to continue, we would have to perturb the array by displacing the vortices in the stream-wise direction. We have not done this because in the actual flow successive pairings are not clearly separated and are random.

A uniform array of vortices can be perturbed in several different ways; for example, by adding a cosine distribution of vorticity to a uniform array, we can enhance the pairing (see Appendix C) and get results similar to the

results presented in this chapter. One could also make the vortices of different strengths or use any combinations of these perturbations.

The perturbation $\beta = 3/16$ (vs. $\beta = 0$ for the unperturbed layer) needed to achieve the observed experimental growth rate of the shear layer may, at first, seem excessive. In the experiments, the downstream vortices exert a significant influence on those in the initial portions of the layer (D&B); recall that the influence of a distant vortical structure on a given point decreases inversely with distance. The cumulative effect of the downstream vortices can be considerable, and, since they tend to be highly turbulent, they may strongly perturb the vortices in the initial section of the mixing layer. Therefore, the value $\beta = 3/16$ may in fact be quite reasonable.

Chapter 6

MIXING LAYER: THREE-DIMENSIONAL COMPUTATIONS

6.1 Preview

In Chapter 5 we started with a two-dimensional initial field, and the numerical simulation of the governing equations stayed two-dimensional. However, actual flows are rarely two-dimensional, and truly turbulent flows are always three-dimensional. (Two-dimensional turbulence is approximated by certain atmospheric structures and in highly stratified fluids.) In this chapter we evaluate the importance of large structures in the development of the mixing layer, which is two-dimensional in the conventional mean sense but contains the three-dimensional structures.

6.2 Boundary Conditions and Mesh-Size Selection

The boundary conditions and coordinate system of Chapter 5 will be used. Periodic boundary conditions will be used in the streamwise (x_1) and spanwise (x_2) directions, and no-stress boundary conditions in the cross-flow (x_3) direction.

The number of meshes used for the cases discussed in this chapter is $16 \times 16 \times 33 = 8448$. The mesh sizes and time step are the same as in the previous chapter. After non-dimensionalizing all coordinates on the initial momentum thickness and the velocity on Δu , the mesh size in the cross-flow (x_3) direction is:

$$h_3 = 1.023$$

In a mixing layer the eddies are suspected of being elongated in the stream-wise direction, so we have set:

$$h_1 = \frac{4}{3} h_3$$

and

$$h_2 = h_3$$

We note that if the mixing layer is completely coherent in the spanwise direction the size of the mesh in this direction (h_2) is not critical.

The non-dimensional time step was set equal to

$$\Delta T = \frac{\Delta u \Delta t}{\theta_{in}} = 0.0799$$

6.3 Initial Conditions

We begin by taking the view that the mixing layer is a superposition of a random velocity ($\tilde{u}$) and a mean velocity profile ($u/\Delta u$). We want the initial random profile to be solenoidal (i.e., $\nabla \cdot \tilde{u} = 0$), random in a region of space (see Fig. 6.1), and to decay to zero outside this region.

In Chapter 3 we showed how to generate an isotropic random velocity field $\underline{u}_I$ on a 16^3 grid. To generate the random part of the initial field that we need here, we start with the field of Chapter 3 and form:

$$\underline{\psi}(I, J, L) = \underline{u}_I(I, J, L-9) \quad L = 14, \dots, 20 \quad (6.1)$$

$$\underline{\psi}(I, J, L) = 0 \quad \text{otherwise}$$

(where I, J, L are the mesh point indices); i.e., a random field over the middle of the shear layer that drops abruptly to zero outside. In order to smooth out the jump between the two regions, $\underline{\psi}$ is filtered in the z -direction with a Gaussian filter. We get:

$$\bar{\underline{\psi}} = \int \underline{\psi}(z') G(z-z') dz' \quad (6.2)$$

where

$$G(z) = \frac{1}{A_3} \exp\left(-\frac{z^2}{6h_3^2}\right)$$

The random portion of the initial field is generated by setting

$$\tilde{\underline{u}} = \nabla \times \bar{\underline{\psi}} \quad (6.3)$$

The initial conditions were completed by adding to $\tilde{u}$ an error function mean velocity:

$$\frac{u}{\Delta u} = 0.5 \operatorname{erf} \frac{(z-L_3/2)}{\sqrt{6} h_3} \quad (6.4)$$

Two cases were run:

$$\text{Case a: } \frac{|\tilde{u}_i|_{\max}}{\Delta u} = 0.01 \quad (i = 1, 2, 3)$$

$$\text{Case b: } \frac{|\tilde{u}_i|_{\max}}{\Delta u} = 0.30 \quad (i = 1, 2, 3)$$

In these two cases the large (grid) structures are assumed to be random fluctuations.

The two-dimensional cases studied in Chapter 5 could be considered as unsteady laminar flows, since there is no randomness. We emphasize that there are at least two kinds of randomness:

- i) Randomness of the pairing in which the vortices vary in shape, separation distance, strength, number, etc., in a random way. In Chapter 5 we computed realizations using spacing as the perturbation.
- ii) Randomness meaning noisy (random) fluctuations.

The calculations described above are designed to look into the second type of randomness. To see what the combined effect would be, we ran still another case in which the initial field contained a vortex pair (with $\beta = 3/16$) and a superimposed random field. For the latter, we took the random field of case (b) described above. This case will be called (c).

Table 6.1 summarizes the cases studied in this chapter. In Appendix D we investigate the interaction between streamwise cellular structures and spanwise vortex pairing.

6.4 Momentum Thickness

In order to study the development of the mixing layer, we would need a measure of the effects of the turbulent rotational region on the non-turbulent irrotational region; the momentum thickness $\theta(t)$ is one such measure. We note that $\theta(t)$, as defined by Eqn. (5.4), is a measure of the momentum defect of the irrotational region. The momentum defect is due to the spreading of vorticity into the irrotational region. Since, in our computation, we have dropped the viscous terms, the growth of the momentum

thickness measures the inviscid mixing or the entrainment of irrotational fluid.

Figure 6.2 shows the non-dimensional momentum thickness θ/θ_{in} (θ_{in} is the initial momentum thickness) plotted vs. T for the three cases considered. We note that in all three cases θ grows linearly with time. The growth rates $(d\theta/\Delta u dt)$ for cases (a) and (b) are not very different, despite the large differences in turbulence levels. The values of 0.008 and 0.011, respectively, are also substantially lower than the growth rate (0.018) reported experimentally by S&J; they are, in fact, lower than any of the values in Table 1.1. The rate of growth of the momentum thickness is only slightly dependent on the intensity of the turbulent fluctuations in cases (a) and (b), and a higher turbulence intensity produces a higher growth rate. Furthermore, when large organized structures are present (case (c)), the momentum thickness growth rate, $d\theta/\Delta u dt = 0.02$ is equal to what it was in the absence of random fluctuations.

Fig. 6.3 shows the non-dimensional momentum thickness θ/θ_{in} plotted vs. T , for case (c) and the two-dimensional case with $\beta = 3/16$. Only at the early stages of the development of the layer do the random fluctuations affect the growth of the momentum thickness.

6.5 Mean Velocity Profiles

An important characteristic of the experimental turbulent mixing layer is the self-similarity of the mean velocity profiles. In our computation, the mean velocity $\langle \bar{u} \rangle_{xy}$ is defined by Eqn. (5.8).

Figures 6.4a, b, and c show $2 \langle \bar{u} \rangle_{xy} / \Delta u$ plotted vs. z/θ at $\Delta T = 2.4$ intervals, for cases (a), (b), and (c), respectively. We obtain self-similar profiles in all cases. This means that self-similarity may be obtained from a wide variety of different flow structures, and does not provide much information about which initial conditions best represent physical reality.

6.6 Mean Turbulence Intensity Profiles

Experimental observations show that the mean turbulence intensity profiles are very nearly self-preserving (Townsend, 1956). This means that:

$$\frac{q^2}{2(\Delta u)^2} = f\left(\frac{z}{\theta}\right) \quad (6.5)$$

Defining the integral of the turbulent energy I_T at a given downstream distance to be

$$I_T = \int_{-\infty}^{+\infty} \frac{q^2}{2(\Delta u)^2} dz \quad (6.6)$$

and substituting (6.5) in (6.6), we get

$$I_T = \theta \int_{-\infty}^{+\infty} f\left(\frac{z}{\theta}\right) d\left(\frac{z}{\theta}\right) = C\theta \quad (6.7)$$

where

$$C = \int_{-\infty}^{+\infty} f(\eta) d(\eta)$$

Non-dimensionalizing on the initial integral of the turbulent energy, $I_{T,in}$, we get

$$\frac{I_T}{I_{T,in}} = \frac{\theta}{\theta_{in}} = \frac{t-t_o}{t_{in}-t_o} \quad (6.8)$$

Equation (6.8) shows that I_T grows linearly with time if the profiles of $q^2/2(\Delta u)^2$ are self-similar. To compute I_T , the mean turbulent energy defined by Eqn. (5.33) was integrated numerically in the z -direction.

Figure 6.5 shows $I_T/I_{T,in}$ plotted vs. T , for the three cases. We note that for all three cases $I_T/I_{T,in}$ decays with time. However, only for case (c), in which large structures are present, did the decay level off.

Figures 6.6a, b, and c show $q^2/2(\Delta u)^2$ plotted vs. z/θ , at $\Delta T = 2.4$ intervals, for cases (a), (b), and (c), respectively. Consistent with the integral of the turbulence energy results, the turbulence intensity decays in time. The most significant drop of the maximum turbulence intensity occurs in the early stages of the development of the layer.

The fact that the integral of the turbulence energy decays, instead of growing linearly with time, is a clear indication that the term (Eqn. (2.16)) used in our equations (2.28) to model the subgrid scale motions,

has too much of an inhibiting effect on the growth of the turbulent fluctuations.

In order to support the above argument, we ran a case in which we started with the same initial conditions as in case (b), but set $C_v = 0$. Fig. 6.7 shows $q^2/2(\Delta u)^2$ plotted vs. z/θ , and Fig. 6.8 shows $I_T/I_{T,in}$ plotted vs. T , for this case. It is clear that the turbulence intensity grows with time, indicating that in case (b) the subgrid scale model is inhibiting the growth of the turbulence energy.

Recall that when the initial conditions contain nothing but large structures we obtain self-similar turbulent intensity profiles (see Section 5.8), even with $C_v = 0.188$. The decay of the total turbulence energy (Fig. 6.5) might suggest that the subgrid scale constant determined for the decay of the isotropic turbulence case might be too high for the mixing layer case. However, the growth rate of the momentum thickness for case (b) is much lower than the growth rate reported experimentally. With $C_v = 0$, the case (b) layer did not grow, i.e., $d\theta/\Delta u dt = 0$, at least up to $T = 9.6$, which indicates that lowering the subgrid scale constant will not give us a momentum growth comparable to the experiments. We thus surmise that it is essential that large structures be included in the initial conditions if the numerical results are to reproduce significant features of the experimental mixing layer. In principle, we could begin with a laminar shear layer and some small perturbations. The Kelvin-Helmholtz instability would then produce large vortical structures and would eventually produce a velocity field with the experimentally observed features. A computation of this type would require at least an order of magnitude more computing time. As we have noted earlier, the subgrid scale model would inhibit the growth of the perturbations and is not adequate for a computation of transitional flow. We shall need to modify the model if transitional flows are to be computed. An alternative approach would be to increase the amplitude of the perturbations and lower the constant of the subgrid scale model, or use a finer mesh.

6.7 Vorticity Contours

In order to investigate the eddy structures and their dynamics, vorticity contours in x - z planes have been plotted in Figs. 6.9 and 6.10, for the three cases considered, at times $T = 0$ and $T = 16.78$.

Figure 6.9a shows the spanwise vorticity contours for case (a), at time $T = 0$. The combination of a weak random velocity field and a smooth mean velocity distribution yields vorticity contours that are almost unaffected by the random fluctuations. The development at $T = 16.78$, shown in Fig. 6.10a, does not indicate any significant effect of the random fluctuations on the mean. The mean field simply masks the weak fluctuations in both the initial conditions and at $T = 16.78$.

Figures 6.9b show the spanwise vorticity contours for case (b), at different spanwise (x - z) planes. The combination of a strong random velocity field and a mean velocity yields vorticity contours that look spotty. At time $T = 16.78$, Figs. 6.10b show that the spots appear much more elongated. At some planes (e.g., plane 5), there are two vortex tubes that appear as if they might pair, while other planes show only one vortex tube. This indicates that the initially strong random fluctuations are being organized by the mean field, and that the layer is developing through a combination of diffusion (due to the subgrid scale model) and vortex pairing.

Figures 6.9c show the spanwise vorticity contours for case (c) at different spanwise (x - z) planes. Adding random fluctuations to the two spanwise vorticities causes the contour lines of the spanwise vorticity to become irregular. At time $T = 16.78$, the vortices have merged in some planes (e.g., planes 1-4) in Figs. 6.10a, whereas in other planes (e.g., planes 5-6) the vortices are still in the process of merging. This indicates that strong random fluctuations can affect the dynamics of vortex pairing.

6.8 Two-Point Correlations

In order to investigate whether or not the mixing layer shows a tendency to increased or decreased spanwise coherence, the spanwise correlation of the streamwise velocity fluctuations ($R_{uu}(r, z)$) was computed. R_{uu} is defined as

$$R_{uu}(r, z) = \frac{\int_x \int_y u''(x, y, z) u''(x, y+r, z) dx dy}{\int_x \int_y u''(x, y, z) u''(x, y, z) dx dy} \quad (6.9)$$

where

$$u'' = \bar{u} - \langle \bar{u} \rangle_{xy}$$

Numerically, this quantity is computed as follows. We first calculate u'' , then take its discrete Fourier transform in the y -direction to yield $\hat{u}''(x, k_2, z)$. $\hat{R}(x, k_2, z)$ is then defined to be equal to

$$\hat{R}(x, k_2, z) = \hat{u}''(x, k_2, z) \hat{u}''^*(x, k_2, z) \quad (6.10)$$

where $\hat{u}''^*$ is the complex conjugate of $\hat{u}''$. Inverse transforming (6.10) yields the discrete equivalent of

$$R(x, r, z) = \int_y u''(x, y, z) u''(x, y+r, z) dy \quad (6.11)$$

Finally, line-averaging (6.11) in the x -direction and normalizing yields the discrete equivalent to (6.9).

Figures 6.11 show R_{uu} at $T = 0$ and $T = 16.78$, plotted vs. r at various z locations. We shall define the correlation length to be the abscissa of the point where R_{uu} first crosses the r -axis.

For case (a), Figs. 6.11a show no significant changes in the correlation length between time $T = 0$ and $T = 16.78$. In some parts of the flow the correlation length seems to increase, whereas in other parts the correlation length seems to decrease. These variations are not significant.

Figures 6.11b show that when we start with a large random initial fluctuation superimposed on a mean profile (case (b)), the correlation length increases with time. This indicates that the layer is becoming more organized in the spanwise direction and is consistent with the result stated earlier that the vorticity tends to clump. Apparently there is a tendency toward the formation of two-dimensional vortices.

Figures 6.11c show that when we add a random field to coherent structures (case (c)), the correlation length decreases slightly with time. The only increase in the correlation length occurs at the center of the layer (plane 17 in our case).

If the spanwise correlation length of the streamwise velocity is taken as a measure of the coherence of the layer, our results tend to indicate that a layer that begins with a random field becomes more coherent, and one that starts with two-dimensional vortical structures loses coherence when the random fluctuations are strong.

6.9 Summary and Conclusions

We have shown that the development of the mixing layer is highly dependent on the initial conditions. This dependence is partly physical and partly numerical. Experimentally, the importance of the initial conditions on the development of the two-dimensional mixing layer has been pointed out by several workers (Bradshaw, 1966; Batt, 1975). Analytically, the subgrid scale models have been developed under the assumption that all the energy transferred by the large resolvable scales to the subgrid scales is dissipated. The decay of the turbulence intensity in cases (a) and (b) indicates that it is doubtful that we can compute transition with the present subgrid scale models. The presence of large structures in the initial conditions is essential to the computation of inhomogeneous turbulent flows.

From the above observations we can conclude that in order to predict the initial development of a shear layer one would need a subgrid scale model that allows the energy of the small scale field to build up and eventually reach equilibrium with the large eddies. However, the later development of a shear layer can be predicted with the present subgrid scale models, provided the large structures are explicitly included in the initial conditions. For other flows, it would appear that inclusion of large structures that at least approximate those of the physical flow is essential to obtain reasonable results. Bass and Orszag (1976) attempted to study the evolution of a passive scalar field in a sheared turbulent velocity field, but were unable to obtain physically realistic results. This may have been due to the omission of the large structures in their initial conditions.

Chapter 7

CONCLUSIONS AND RECOMMENDATIONS

In this work we have developed an approach to three-dimensional, time-dependent computations of flows using the vorticity equations. A general method of deriving conservation properties that is applicable to any numerical method in incompressible fluid mechanics was given; its use simplifies the analysis of numerical schemes.

The use of a filter which is smooth in real space has been shown to be essential for the treatment of rotational-irrotational region interactions. The use of Fourier transform methods allows accurate and fast treatment of the term $\overline{u_j \omega_i} - \overline{u_i \omega_j}$, which arises as a consequence of filtering. This is a definite improvement over the expansion in Taylor series (Leonard, 1973) used in previous studies (Kwak et al., 1975), which we believe should be used only when the use of transform methods is not justifiable.

The vorticity equations have been shown to provide a satisfactory basis for the simulation of homogeneous isotropic turbulence. Comparison of our results with results obtained using the primitive variable equations (Mansour et al., 1977; Moin et al., 1978) shows no significant differences.

A new subgrid scale model has been developed and shown to give results comparable to those obtained using the vorticity model (Kwak et al., 1975). The new model offers advantages both in computational speed and in storage. We found that, for the calculation of isotropic homogeneous turbulence, the subgrid scale constant depends only slightly on the numerical method used. The variation is about ten percent and is not likely to have a significant effect on the computed results in shear flows. The use of Fourier spatial differencing has allowed us to look more carefully at the subgrid scale model, and it has been found that replacing exact derivatives with second-order differences (roughly equivalent to averaging the model spatially (Love and Leslie, 1977)) produces improved behavior of the spectrum.

No-stress boundary conditions in one direction and periodic boundary conditions in the other two directions have been incorporated in a three-dimensional, time-dependent code. Flows in which these boundary conditions

can be justified (e.g., two-dimensional wakes, planar jets, mixing layer) can be investigated using this code. We chose the mixing layer.

Two-dimensional computations of the turbulent mixing layer have shown that pairing vortices produce self-similar mean velocity and turbulence intensity profiles. The growth rate of the layer is strongly dependent on the initial conditions, a fact also observed experimentally.

Three-dimensional computations have shown that the presence of large, organized (i.e., not random) structures is essential if the simulation is to reproduce the essential features observed in the experiments. These computations suggest that in order to simulate the initial development of a shear layer one would need a subgrid scale model that allows the energy of the small scale field to build up with time and eventually reach equilibrium with the large eddies. However, the later development of a shear layer can be predicted with the present subgrid scale models, provided the large structures are explicitly included in the initial conditions.

Our results using different initial conditions indicate that self-similarity of the mean velocity profiles can be obtained more easily than self-similarity of the turbulence intensities. The addition of strong random fluctuations to a flow containing pairing vortices disturbs the pairing in a way that causes the vortex tubes to exhibit spanwise variations, and whether or not the merging is completed depends on the spanwise locations. This may explain the onset of three-dimensionality seen in experiments. Fig. 7.1 is a conjecture of what we think might happen. The section of the vortex tube that did not merge could interact with the vortex structure just ahead (or just behind) to form a horseshoe vortex. This horseshoe vortex may get stretched over several rollers, giving the appearance of cellular structures (B&R, Konrad).

In Appendix D we study the interaction between streamwise and spanwise vorticity. Again, the detailed results depend strongly on the initial conditions. Each free shear flow is unique, and the universality that is sought exists only at large downstream distances. This may mean that the computational "prediction" of free shear flows is feasible only to moderate accuracy; the precise behavior of an individual free shear flow may depend on physical details that are not easily controlled. This means that some experimentation will always be necessary.

Work remains to be done on the development of a subgrid scale model that incorporates flow-regime dependence. Ideally, one would like a model that can handle both transition and developed turbulence. With such a model, problems associated with the initial conditions can be studied more carefully, since the linear stability theory is well understood and the initial conditions can be chosen to be solutions of the Orr-Sommerfeld equations. This kind of computation will help understand the effect of the initial conditions on the development of the mixing layer, but will not reproduce experiments exactly.

In the case of the mixing layer, the use of periodic boundary conditions is justifiable only if we move with the mean speed of the flow. However, the size of the eddies grows linearly with the streamwise distance (in our frame linearly in time), and we reach a point at which the size of the box must be increased. In a stationary frame this problem can be avoided, but inflow-outflow boundary conditions must be used. We suggest that future work should concentrate on developing a method of treating the inflow-outflow boundary conditions.

Eventually, it may be possible to treat practical flows such as airfoils, combustion chambers, etc., by these methods. Before that can be done, much more effort should first be devoted to developing subgrid scale models, treatment of boundary conditions, mesh layout and/or mapping, numerical methods, filters, etc., which are the important building blocks of large-eddy simulation.

References

1. Arakawa, A. (1966), "Computational Design for Long-Term Numerical Integration of the Equations of Fluid Motion: Two-Dimensional Incompressible Flow. Part I," Jour. of Computational Physics, 1, No. 1, 119-143.
2. Ashurt, W. T. (1977), "Numerical Simulation of Turbulent Mixing Layers via Vortex Dynamics," Proceedings of Symposium on Turbulent Shear Flows, Penn. State Univ.
3. Bass, A., and Orszag, S. A. (1976), "Spectral Modeling of Atmospheric Flows and Turbulent Diffusion," EPA-600/4-76-007.
4. Batt, R. G. (1975), "Some Measurements on the Effect of Tripping the Two-Dimensional Shear Layer," AIAA J., 13, No. 2, Tech. Notes, 245-247.
5. Brown, G. L., and Roshko, A. (1974), "On Density Effects and Large Structures in Turbulent Mixing Layers," J. Fluid Mech., 64, 775-816.
6. Champagne, F. H., Pao, Y. H., and Wygnanski, I. J. (1976), "On the Two-Dimensional Mixing Regions," J. Fluid Mech., 74, Part 2, 209-250.
7. Chandrsuda, C., and Bradshaw, P. (1975), "An Assessment of the Evidence for Orderly Structure in Turbulent Mixing Layers," I.C. Aero Report 75-03.
8. Clark, R. A., Ferziger, J. H., and Reynolds, W. C. (1977), "Evaluation of Subgrid-Scale Turbulence Models Using a Fully Simulated Turbulent Flow," Report TF-9, Mechanical Engrg. Dept., Stanford University.
9. Comte-Bellot, G., and Corrsin, S. (1971), "Simple Eulerian Time Correlation of Full and Narrow-Band Signals in Grid-Generated Isotropic Turbulence," J. Fluid Mech., 48, Part 2, 273-337.
10. Cooley, J. W., and Tukey, J. W. (1965), "An Algorithm for the Machine Calculation of Complex Fourier Series," Math. Comput., 19, 90, 297-301.
11. Deardorff, J. W. (1970), "A Numerical Study of Three-Dimensional Turbulent Channel Flow at Large Reynolds Numbers," J. Fluid Mech., 41, Part 2, 453-480.
12. Dimotakis, P. E., and Brown, G. L. (1976), "The Mixing Layer at High Reynolds Number: Large-Structure Dynamics and Entrainment," J. Fluid Mech., 78, Part 3, 535-560.
13. Donaldson, C. duP. (1972), "Calculation of Turbulent Shear Flows for Atmospheric and Vortex Motions, AIAA J., 10, No. 1, 4-12.
14. Fiedler, H., and Thies, H. J. (1977), "Some Observations in a Large, Two-Dimensional Shear Layer," Proceedings of Symposium on Turbulence, Berlin; Springer.

15. Foss, J. F. (1977), "The Effects of the Laminar/Turbulent Boundary Layer States on the Development of a Plane Mixing Layer," Proceedings of Symposium on Turbulent Shear Flows, Penn. State Univ., 11.33-11.42.
16. Fox, O. G., and Orszag, S. A. (1973), "Pseudo-spectral Approximation to Two-Dimensional Turbulence," J. of Comp. Physics, 11, No. 4, 612-619.
17. Konrad, J. H. (1976), "An Experimental Investigation of Mixing in Two-Dimensional Turbulent Shear Flows with Applications to Diffusion-Limited Chemical Reactions," Project SQUID, Tech. Rept. CIT-8-PU.
18. Kreiss, H. O., and Oliger, S. (1973), Methods for the Approximate Solution of Time-Dependent Problems, GARP Publications, Series No. 10, WMO, Geneva.
19. Kwak, D., Reynolds, W. C., and Ferziger, J. H. (1975), "Three-Dimensional, Time-Dependent Computation of Turbulent Flow," Report No. TF-5, Mech. Engrg. Dept., Stanford University.
20. Lamb, H. (1932), Hydrodynamics, Dover Pub., N. Y.
21. Lanczos, C. (1956), Applied Analysis, Prentice-Hall Pub. Co., Englewood, N. J.
22. Leonard, A. (1974), "On the Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows," Adv. in Geophysics, 18A, 237.
23. Liepmann, H. W., and Laufer, J. (1947), "Investigations of Free Turbulent Mixing," NACA Tech. Note No. 1257.
24. Lilly, D. K. (1965), "On the Computational Stability of Numerical Solutions of Time-Dependent, Nonlinear Geophysical, Fluid Dynamic Problems," Monthly Weather Review, 93, No. 1, 11-26.
25. Love, M. D., and Leslie, D. C. (1977), "Studies of Subgrid Modeling with Classical Closures and Bergers' Equation," Proc. of Symposium on Turbulent Shear Flows, Penn. State Univ.
26. Mansour, N. N., Moin, P., Reynolds, W. C., and Ferziger, J. H. (1977), "Improved Methods for Large-Eddy Simulations of Turbulence," Proc. of Symposium on Turbulent Shear Flows, Penn. State Univ.
27. Moin, P., Mansour, N. N., Mehta, U. B., Ferziger, J. H., and Reynolds, W. C. (1978), "Improvements in Large-Eddy Simulation Technique: Special Methods and High -Order Statistics," Report TF-10, Mech. Engrg. Dept., Stanford University.
28. Moin, P., Reynolds, W. C., and Ferziger, J. H. (1978), "Large-Eddy Simulation of an Incompressible Turbulent Channel Flow," Rept. TF-12, Mech. Engrg. Dept., Stanford University.
29. Moore, D. W., and Saffman, P. G. (1975), "The Density of Organized Vortices in a Turbulent Mixing Layer," J. Fluid Mech., 69, Part 3, 465-473.

30. Murthy, J. N. B. (1975), "Turbulent Mixing in Non-Reactive and Reactive Flows - a Review," Proceedings of SQUID Workshop on Turbulent Mixing in Non-Reactive and Reactive Flows, Purdue Univ.
31. Orszag, S. A. (1973), "Comparison of Pseudo-Spectral and Spectral Approximation," Studies in Applied Mathematics, LI, No. 3, 253-259.
32. Orszag, S. A., and Israeli, M. (1974), "Numerical Simulation of Viscous Incompressible Flows," Annual Review of Fluid Mechanics, 6, 281-318.
33. Orszag, S. A., and Pao, Y. H. (1974), "Numerical Computation of Turbulent Shear Flows," Advances in Geophysics, 18A, Academic Press, Inc., N. Y. C., 225-236.
34. Oster, D., Wgnanski, I. J., and Fiedler, H. (1976), "Some Preliminary Observations on the Effect of Initial Conditions on the Structure of the Two-Dimensional Turbulent Mixing Layer," Proc. of SQUID Conference, Plenum Press.
35. Oster, D., Dziomba, B., Wgnanski, I., and Fiedler, H. (1977), "On the Effect of Initial Conditions on the Two-Dimensional Turbulent Mixing Layer," Proceedings of Symposium on Turbulence, Berlin; Springer.
36. Patel, R. P. (1973), "An Experimental Study of a Plane Mixing Layer," AIAA J, 11, 1, 67-71.
37. Patnaik, P. G., Sherman, F. S., and Corcos, G. H. (1976), "A Numerical Simulation of Kelvin-Helmholtz Waves of Finite Amplitude," J. Fluid Mech., 73, Part 2, 215-240.
38. Phillips, N. A. (1959), "An Example of Nonlinear Computational Instability," The Atmosphere and Sea in Motion, Rockefeller Inst. Press, N. Y., 501-504.
39. Roshko, A. (1976), "Structure of Turbulent Shear Flows: A New Look," AIAA J., 14, No. 10, 1349-1357.
40. Schumann, U. (1973), "A Procedure for the Direct Numerical Simulation of Turbulence Flows in Plate and Annular Channels and Its Application in the Development of Turbulence Models," Dissertation, Univ. of Karlsruhe, NASA Tech. Translation NASA TT T 15, 391.
41. Shanaa, S., Ferziger, J. H., and Reynolds, W. C. (1975), "Numerical Simulation of Turbulence in the Presence of Shear," Report TF-6, Mech. Engrg. Dept., Stanford University.
42. Singleton, R. C. (1967), "On Computing the Fast Fourier Transforms," Communications of the ACM, 10, No. 10, 647-654.
43. Smagorinsky, J. (1963), "General Circulation Experiments with the Primitive Equations. I. The Basic Experiment," Monthly Weather Review, 91, No. 3, 99-165.

44. Spencer, B. W., and Jones, B. G. (1971), "Statistical Investigation of Pressure and Velocity Fields in the Turbulent Two-Stream Mixing Layer," AIAA Paper #71-613.
45. Tennekes, H., and Lumley, J. L. (1974), A First Course in Turbulence, MIT Press, Cambridge, Mass.
46. Thies, H. J. (1977), "Experimentelle Untersuchung der freien Scherschicht mit gestörten Angangsbedingungen," Diplomarbeit Tu - Berlin, FB 9.
47. Townsend, A. A., The Structure of Turbulent Shear Flow, Cambridge Univ. Press, Cambridge, England.
48. Wang, S. S. (1977), "A Study of the Kelvin-Helmholtz Instability by Grid-Insensitive Vortex Tracing," Ph.D. thesis, Stanford University, Institute of Plasma Research Report No. 710.
49. Winant, C. D., and Browand, F. K. (1974), "Vortex Pairing: the Mechanism of Turbulent Mixing-Layer Growth at Moderate Reynolds Number," J. Fluid Mech., 63, Part 2, 237-255.
50. Wygnanski, I., and Fiedler, H. E. (1970), "The Two-Dimensional Mixing Region," J. Fluid Mech., 41, Part 2, 327-361.
51. Yule, A. J. (1971), "Spreading of Turbulent Mixing Layers," AIAA J., 10, No. 5, Tech. Notes, 686-687.

Table 1.1

EXPERIMENTAL RESULTS

(Table from Fiedler and Thies, 1977)

Author(s)	u_2/u_1	Re_L^*	L^* (mm)	σ_o	$\frac{d\theta}{\Delta u dt}$	Remarks
Liepmann & Laufer (1947)	0	$9 \cdot 10^5$	900	11.76	0.016	No trip
Wynanski & Fiedler (1970)	0	$5 \cdot 10^5$	600	8.70	0.022	Trip
Batt (1975)	0	$7 \cdot 10^5$	640	8.89	0.022	Trip
	0	$7 \cdot 10^5$	640	11.76	0.016	No trip
Spencer & Jones (1971)	0	$1 \cdot 10^6$	560	10.52	0.018	No trip
Champagne, Pao & Wynanski (1976)	0	$4 \cdot 10^5$	600	9.62	0.020	Trip (B.L. not turb.)
Patel (1973)	0	$2 \cdot 10^6$	1020	10.53	0.018	No trip
Oster, Wynanski & Fiedler (1976)	0	$1.1 \cdot 10^6$	1100	9.21	0.021	Trip
	0	$1.1 \cdot 10^6$	1100	11.29	0.017	No trip
Foss (1977)	0	$6.7 \cdot 10^5$	510	9.00	0.021	Turb. B.L.
	0	$6.7 \cdot 10^5$	510	12.12	0.016	Lam. B.L.
Dimotakis & Brown (1976)	0.2	$3 \cdot 10^5$	600	9.87	0.020	No trip
Oster, Wynanski & Fiedler (1976)	0.4	$2.8 \cdot 10^5$	470	12.12	0.016	Trip
	0.4	$2.8 \cdot 10^5$	470	10.81	0.018	No trip
Spencer & Jones (1971)	0.3	$1 \cdot 10^6$	680	12.31	0.016	No trip
	0.6	$2.8 \cdot 10^5$	320	13.14	0.015	No trip
Yule (1971)	0.3	$5 \cdot 10^5$	650	9.44	0.020	No trip
	0.61	$1.4 \cdot 10^5$	290	9.23	0.021	No trip
						No B.L.-suction
Thies (1977)	0	$2.4 \cdot 10^6$	3600	10.05	0.019	No trip
		$3.8 \cdot 10^6$		9.52	0.020	
		$5.1 \cdot 10^6$		9.09	0.021	
		$2.4 \cdot 10^6$		10.31	0.019	2 mm trip
		$4.2 \cdot 10^6$		9.37	0.021	
		$2.4 \cdot 10^6$		10.24	0.019	4 mm trip
		$3.7 \cdot 10^6$		9.57	0.020	
		$5.1 \cdot 10^6$		9.15	0.021	
		$2.4 \cdot 10^6$		10.23	0.019	Zig-zag trip

Table 1.1 (cont.)

Author(s)	u_2/u_1	Re_L^*	L^* (mm)	σ_o	$\frac{d\theta}{\Delta u dt}$	Remarks
Thies (1977) (cont.)	0	$8.0 \cdot 10^6$	3600	9.17	0.021	B.L.-suction
		$2.5 \cdot 10^6$		10.10	0.019	No trip
		$0.8 \cdot 10^6$		13.13	0.015	("near" re-
		$2.4 \cdot 10^6$		9.80	0.020	gion)
		$0.8 \cdot 10^6$		9.43	0.020	2 mm trip
		$8.0 \cdot 10^6$		8.95	0.022	("near re-
		$2.0 \cdot 10^6$		9.6	0.020	gion)
		$8.0 \cdot 10^6$		9.0	0.021	4 mm trip
		$2.4 \cdot 10^6$		10.21	0.019	8 mm trip
						Zig-zag trip

The following assumptions were used to reduce the data:

$$\sigma_o = \sigma \frac{\Delta u}{\Sigma u}$$

$$L^* = L \frac{\Delta u}{\Sigma u}$$

$$Re_L^* = \frac{\Delta u L^*}{\nu}$$

$$L = x_{\max}$$

$$\sigma = \frac{2}{\eta_{0.1} - \eta_{0.95}}$$

$$\Delta u = u_1 - u_2$$

$$\Sigma u = u_1 + u_2$$

$$\frac{d\theta}{\Delta u dt} = \frac{1}{2.07 \sigma_o \sqrt{2\pi}}$$

— Boundary layer turbulent at the splitter plate
 - - - - Boundary layer laminar at the splitter plate

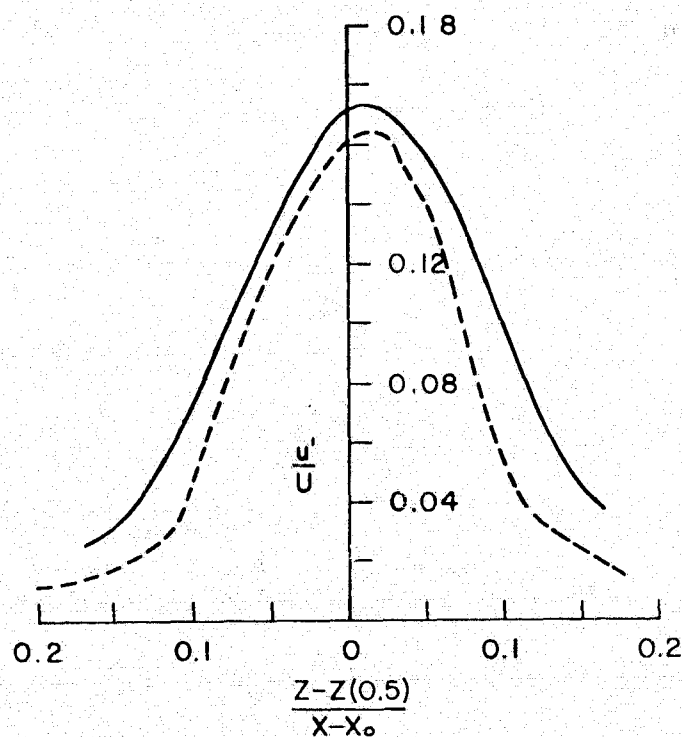


Fig. 1.1. r.m.s. streamwise velocity profiles for different initial conditions (experimental results from Foss, 1977).

— Boundary layer turbulent at the splitter plate
 - - - Boundary layer laminar at the splitter plate

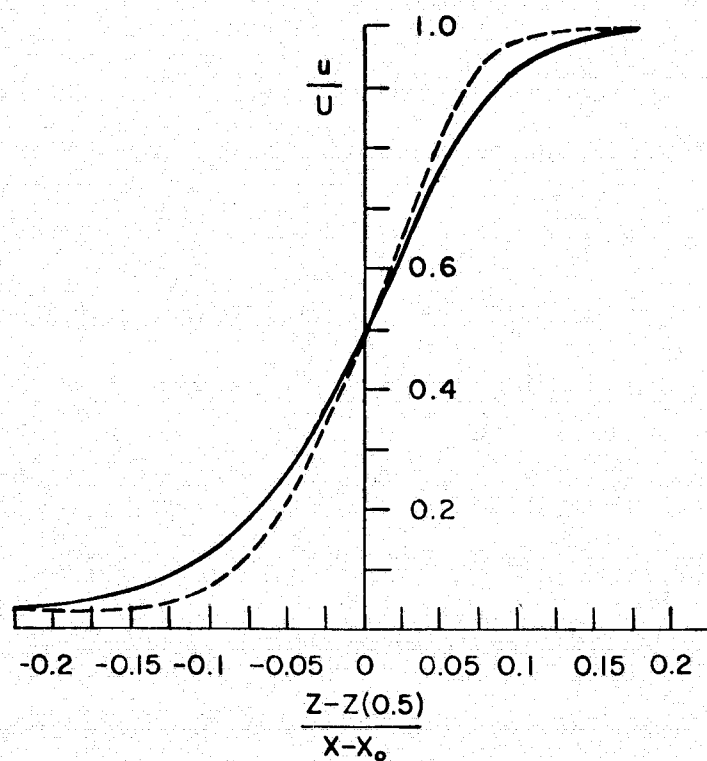


Fig. 1.2. Mean velocity profiles for different initial conditions (experimental results from Foss, 1977).

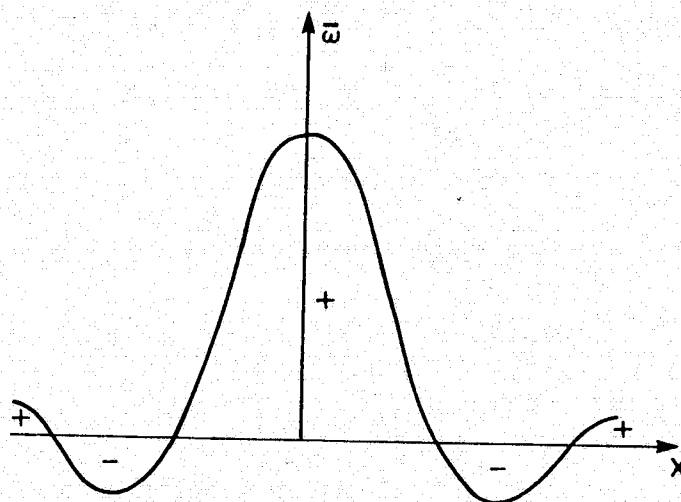


Fig. 2.1. Filtered point vortex with an SCK (sharp cut-off in k -space) filter, $y = 0$.

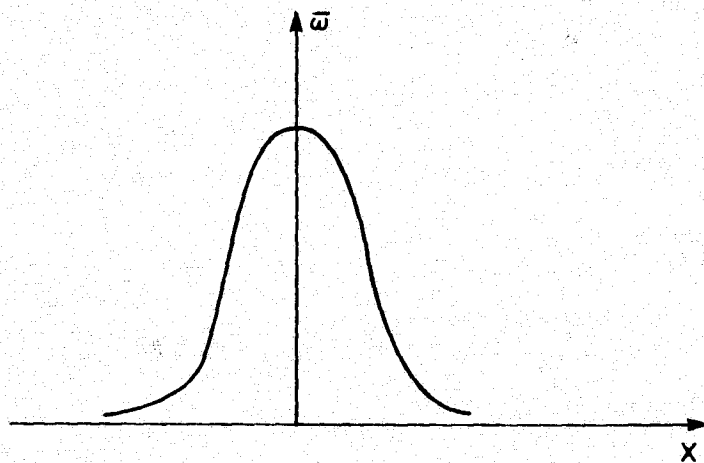


Fig. 2.2. Filtered point vortex with a GS (Gaussian in real space) filter, $y = 0$.

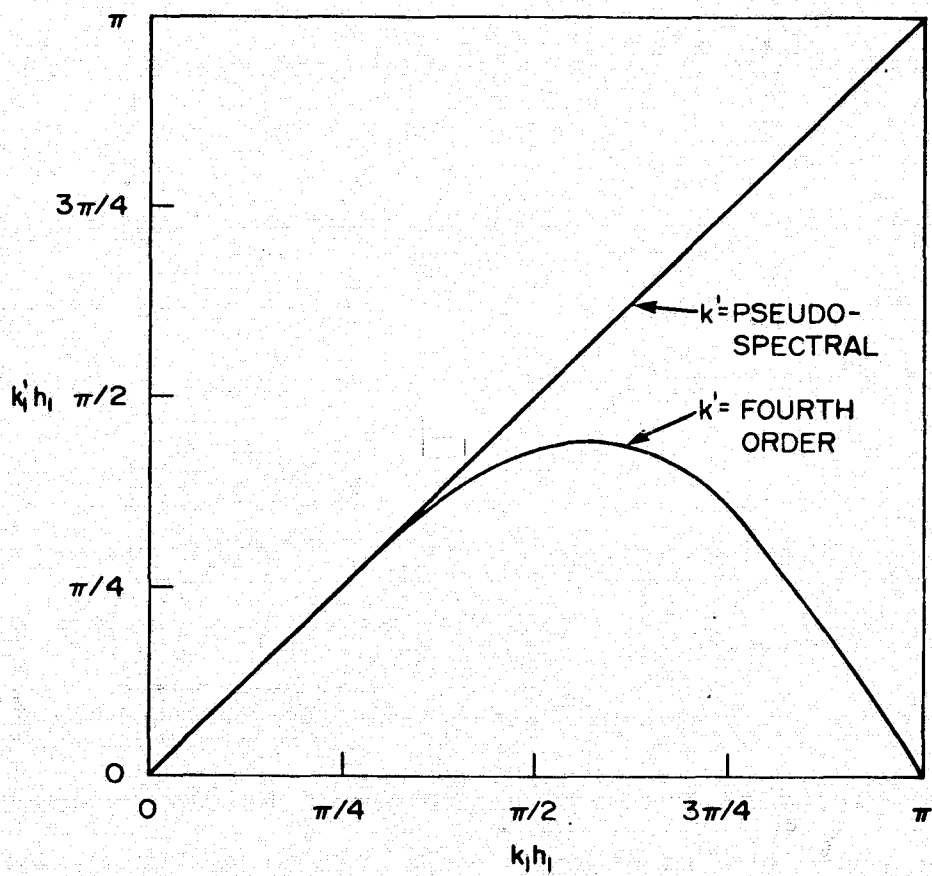


Fig. 3.1. Comparison of modified wave numbers.

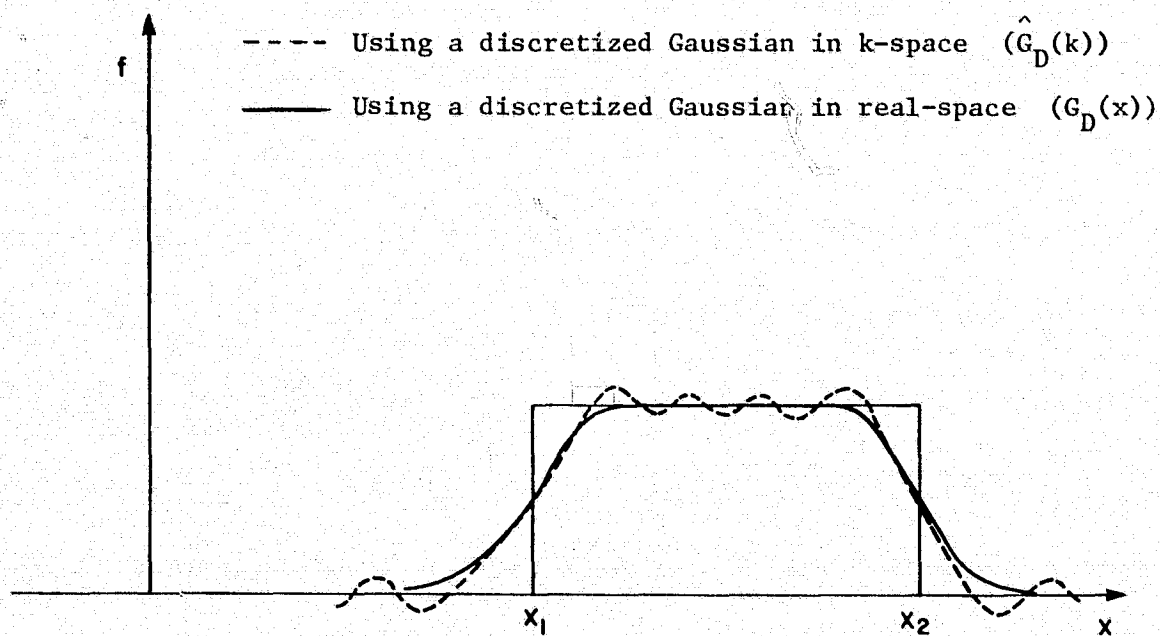


Fig. 3.2. Filtered top-hat function.

Table 4.1

Computations of the Decay of Isotropic Turbulence

No. of Mesh Points	Subgrid Scale Model	Numerical Scheme	Numerical Scheme for the Subgrid Scale Model	Model Constant	Figure
$16 \times 16 \times 16$	Model ω -1	Fourth-order diff.	Second-order diff.	$C_v = 0.235$	4.2
$16 \times 16 \times 16$	Model ω -1	Pseudo-spectral	Pseudo-spectral	$C_v = 0.212$	4.3
$16 \times 16 \times 16$	Model ω -1	Pseudo-spectral	Second-order diff.	$C_v = 0.213$	4.4
$16 \times 16 \times 16$	Model ω -2	Pseudo-spectral	Pseudo-spectral	$C_v = 0.186$	4.5
$16 \times 16 \times 16$	Model ω -2	Pseudo-spectral	Second-order diff.	$C_v = 0.188$	4.6
$32 \times 32 \times 32$	Model ω -2	Pseudo-spectral	Second-order diff.	$C_v = 0.188$	4.7

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

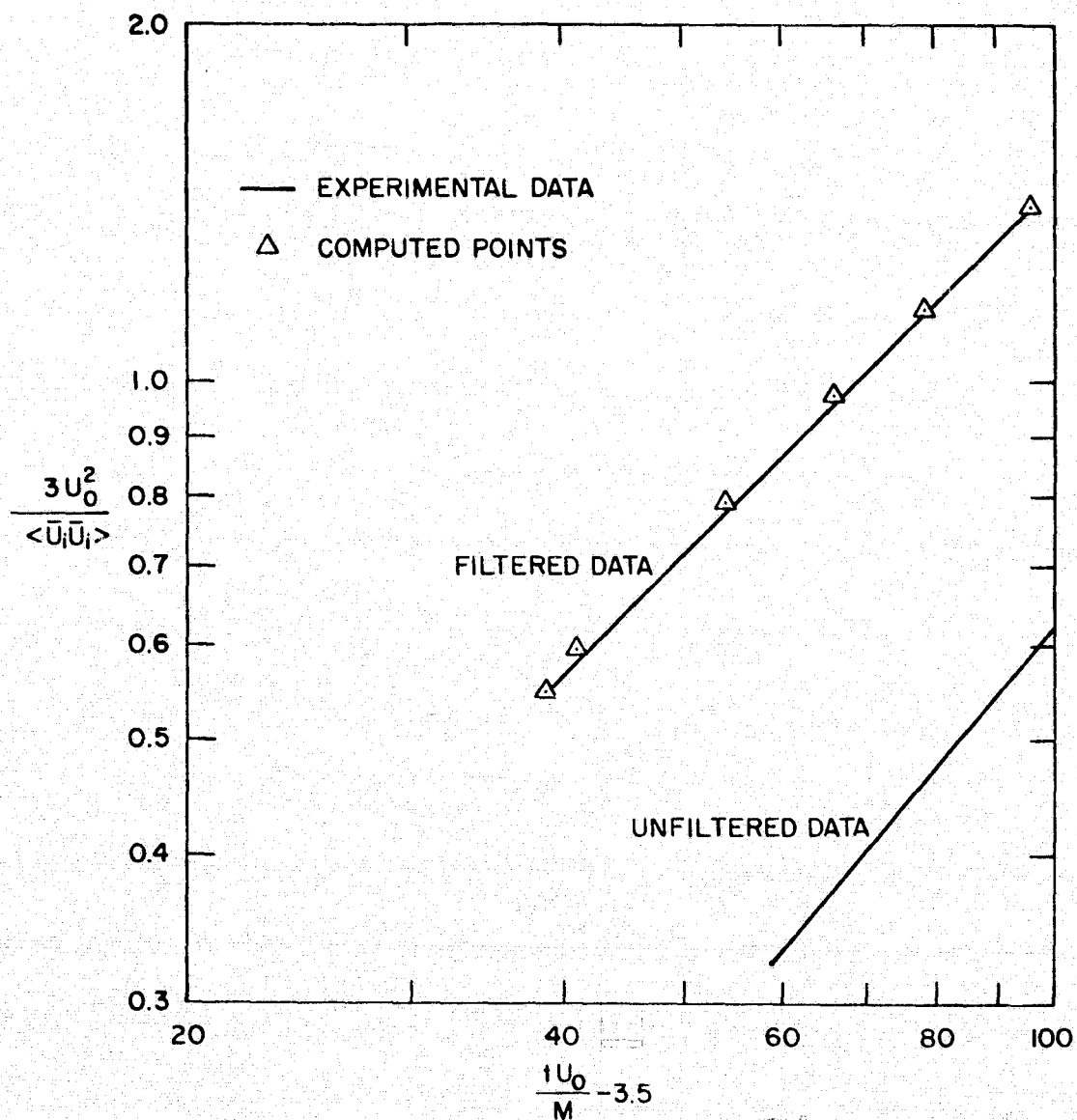


Fig. 4.1. Decay of mean square filtered velocity for $16 \times 16 \times 16$ mesh. $\langle \rangle$ = average over all space.

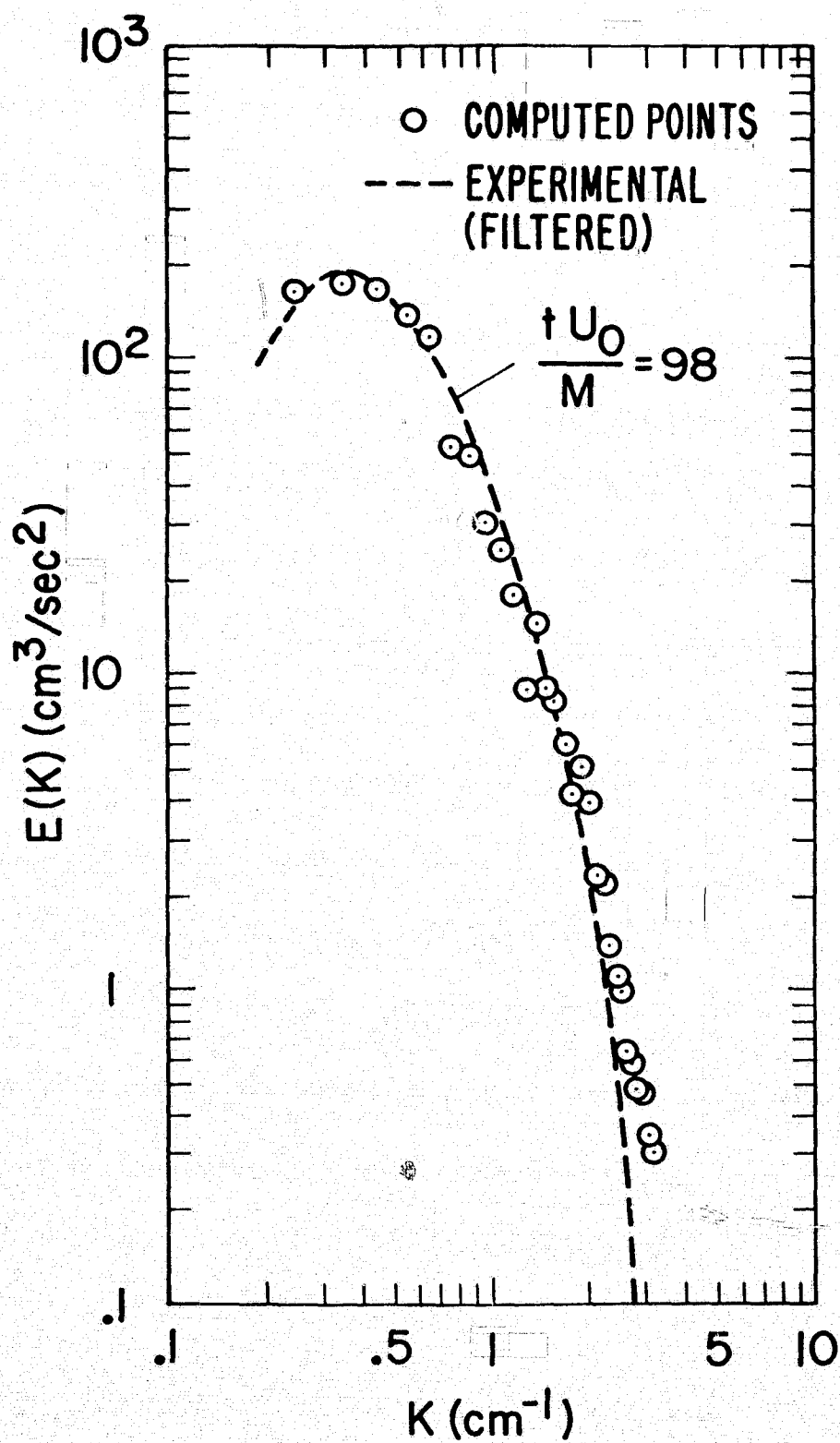


Fig. 4.2. Filtered energy spectra. Fourth-order differencing with 16^3 mesh; model ω -1, $C_v = 0.235$.

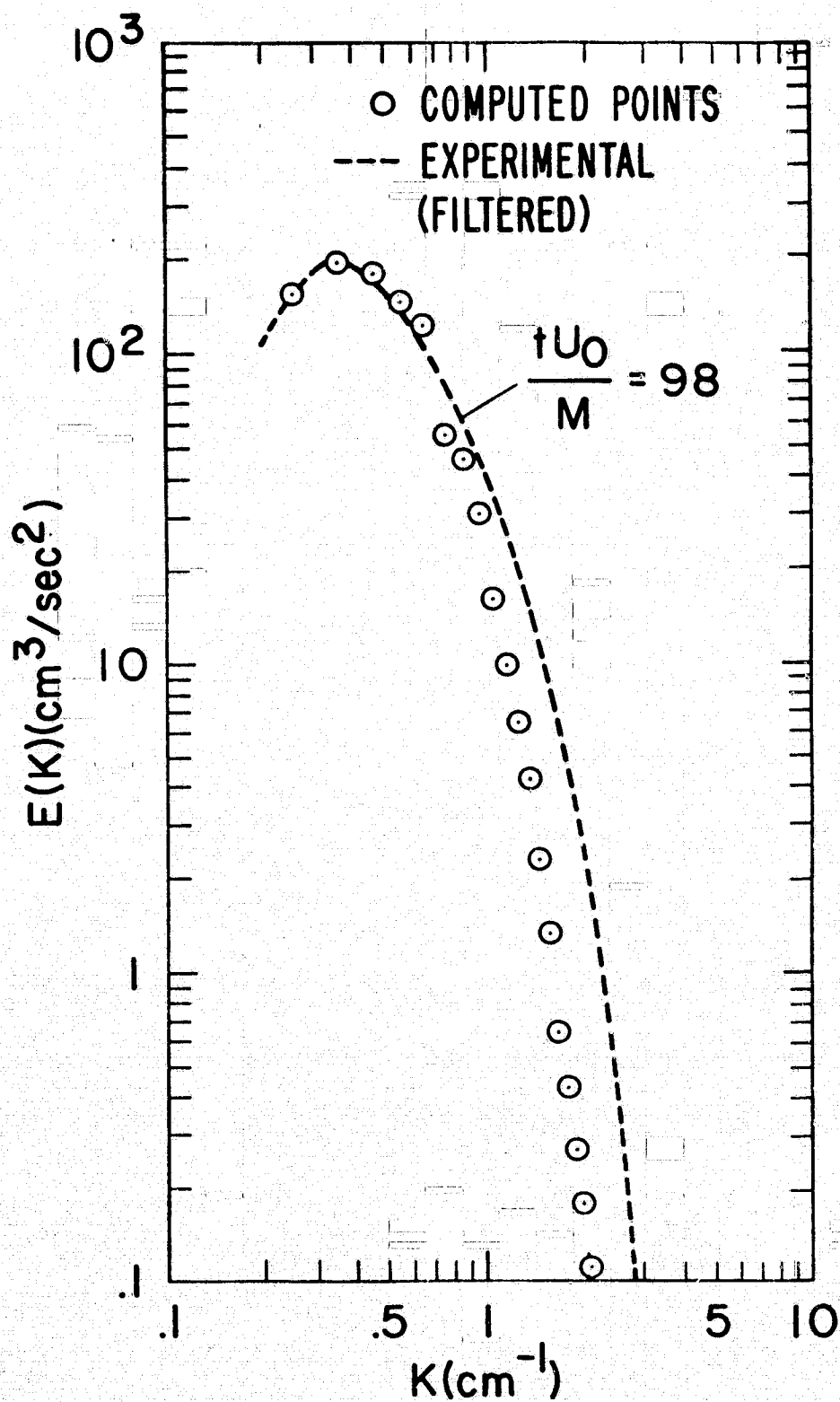


Fig. 4.3. Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -1. $C_v = 0.212$.

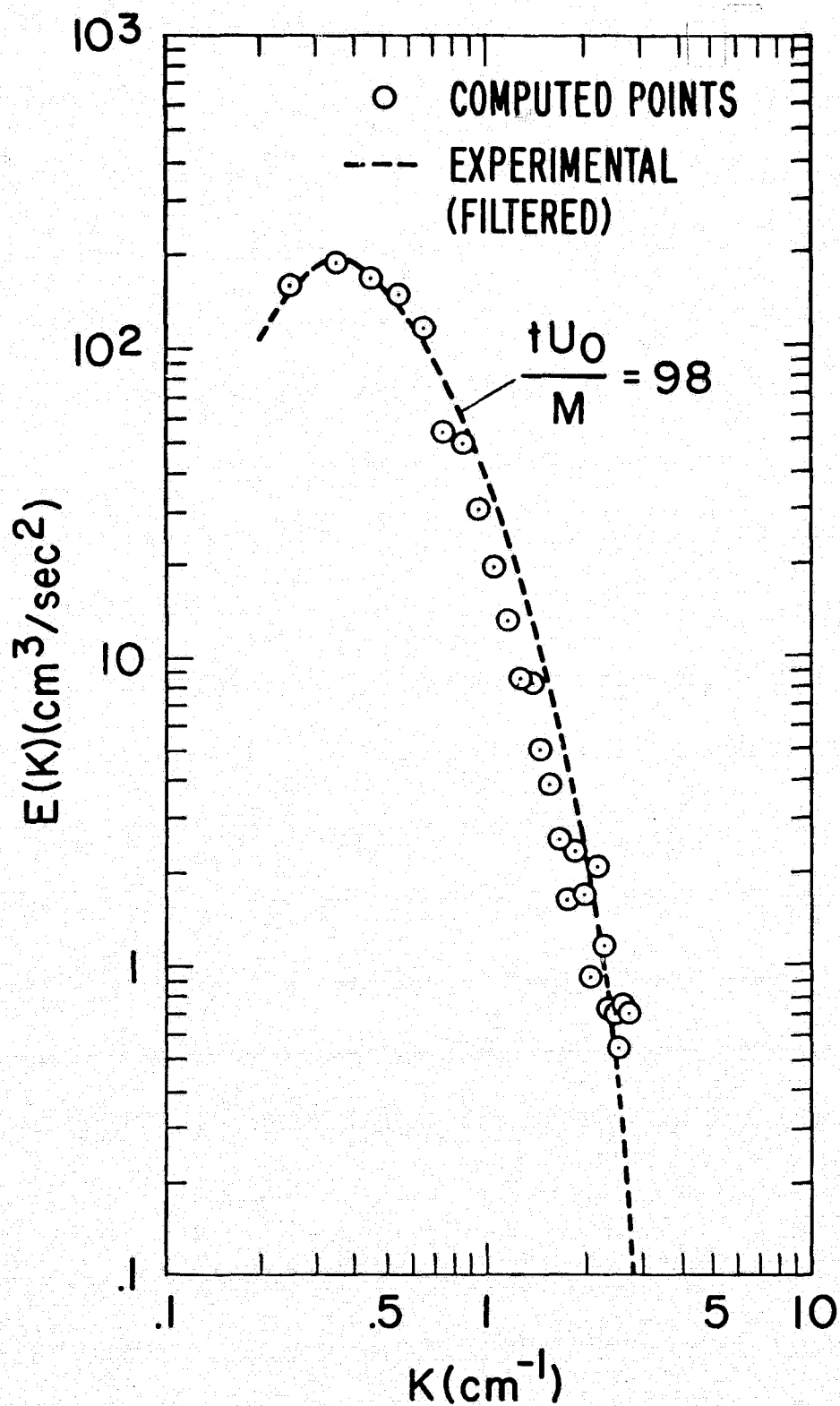


Fig. 4.4. Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; 2nd-order differencing for model $\omega=1$. $C_v = 0.213$.

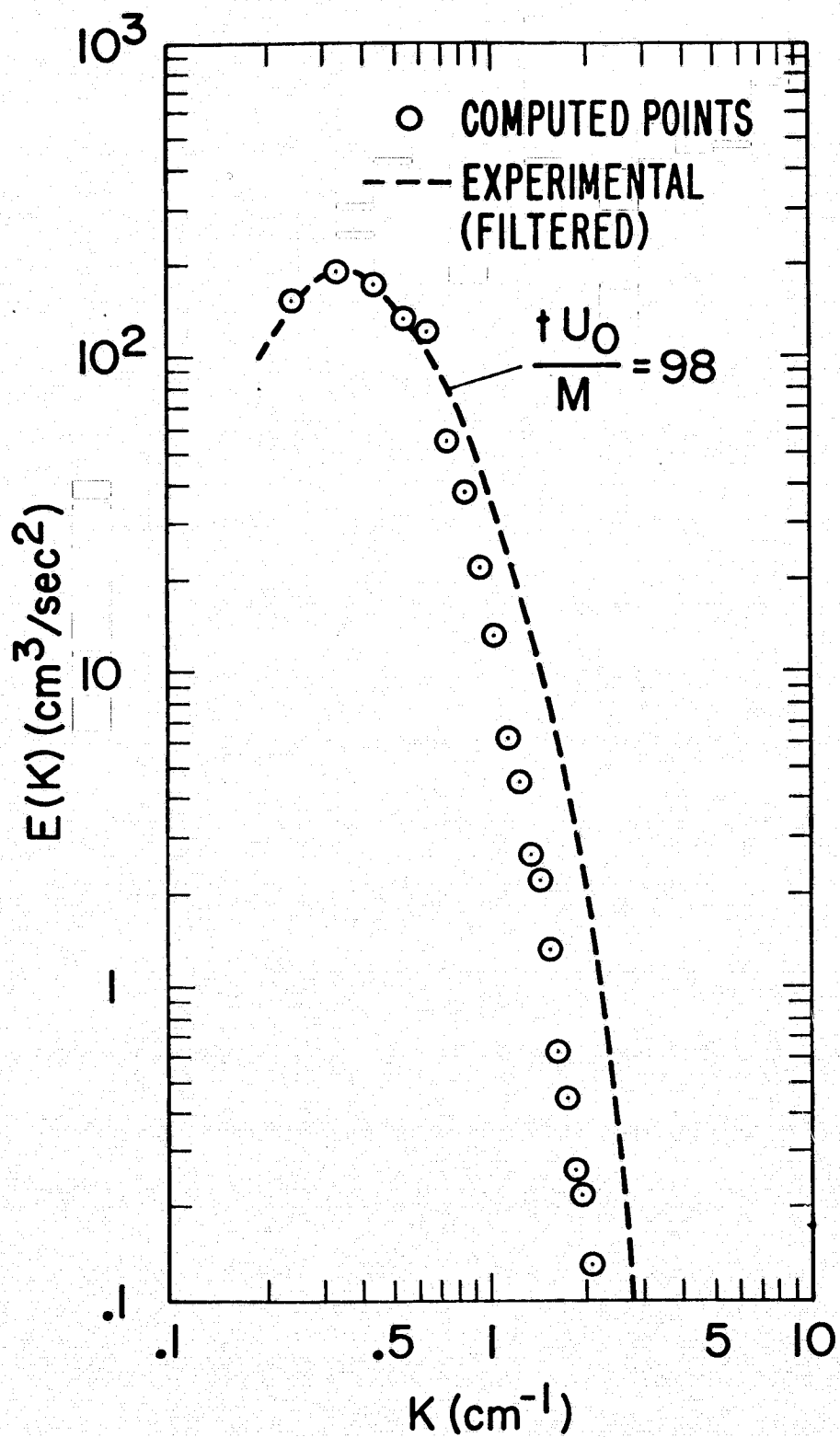


Fig. 4.5. Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; model ω -2. $C_v = 0.186$.

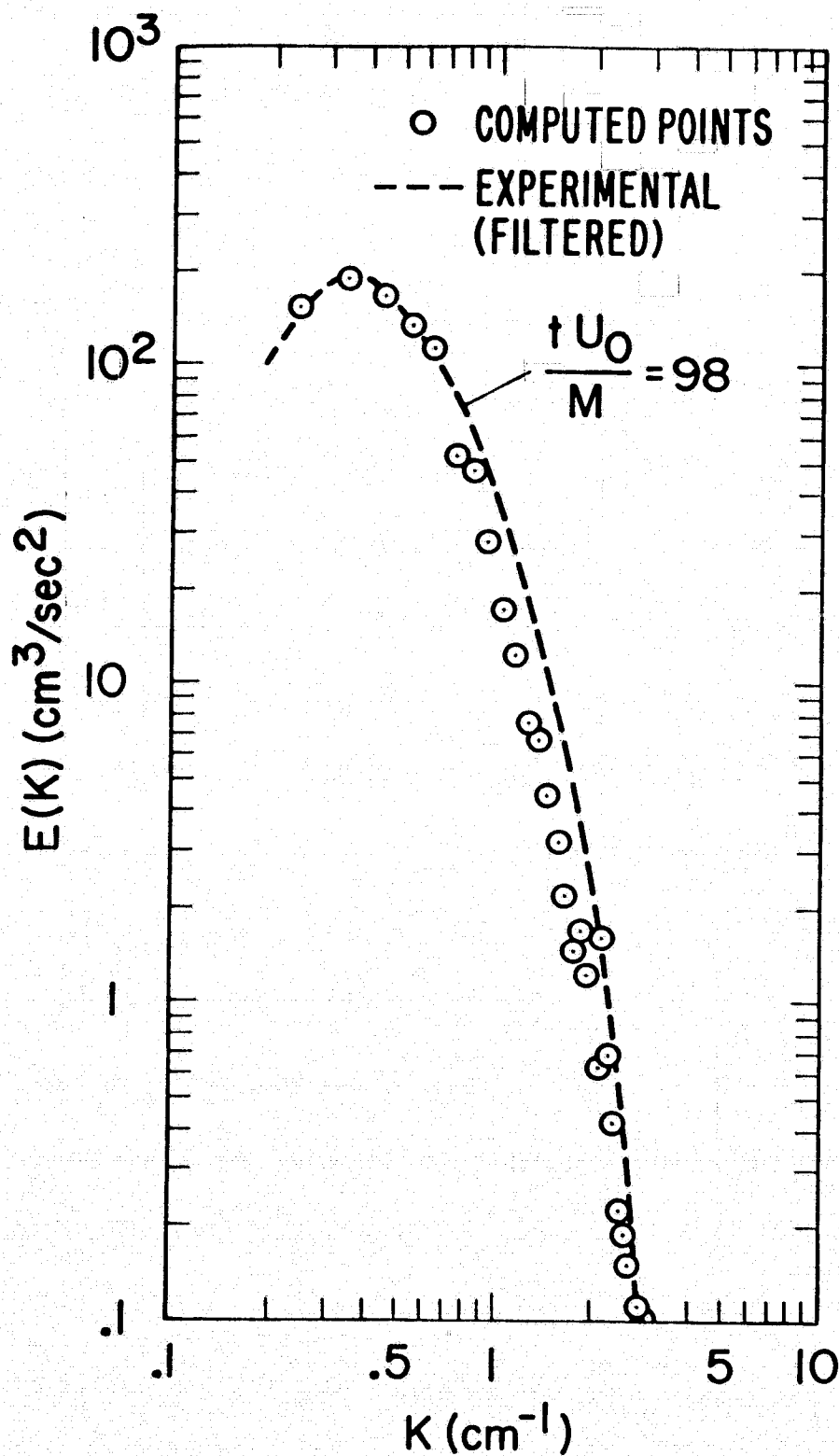


Fig. 4.6. Filtered energy spectra. Pseudo-spectral computation with 16^3 mesh; 2nd-order differencing for model ω -2. $C_v = 0.188$.

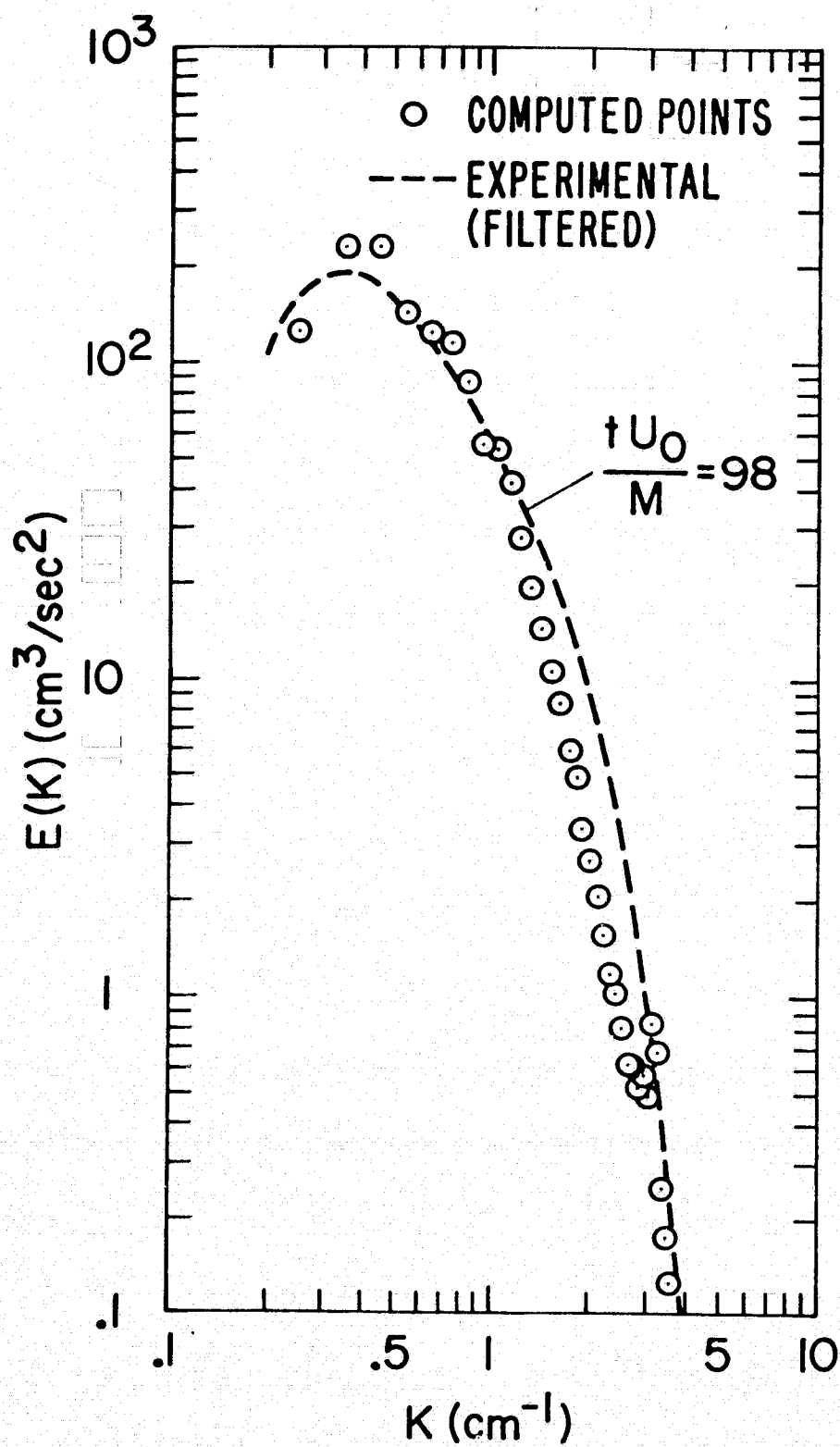


Fig. 4.7. Filtered energy spectra. Pseudo-spectral computation with 32^3 mesh; 2nd-order differencing model ω -2. $C_v = 0.188$.

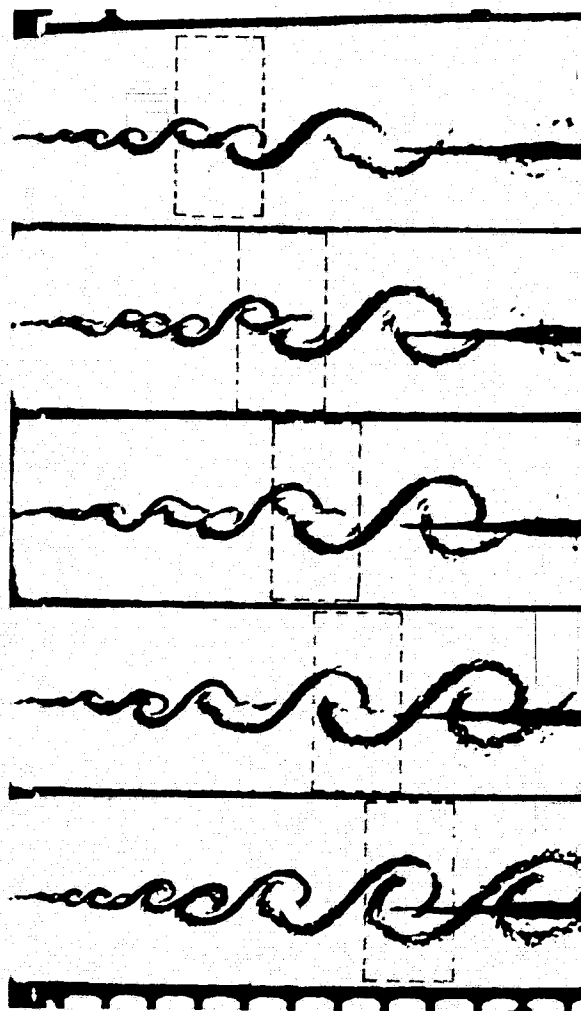


Fig. 5.1. Coherent structure in a mixing layer (Roshko, 1976). Dashed box: schematic of a computational box that moves approximately with the mean velocity.

C-2

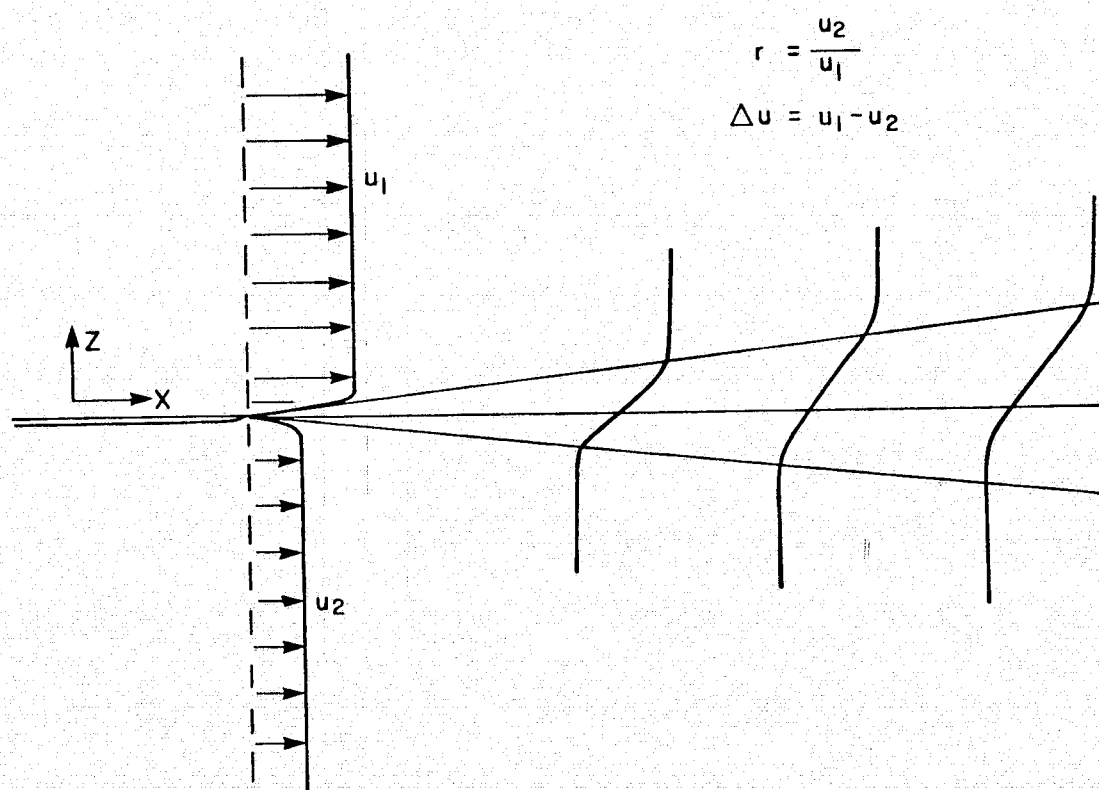


Fig. 5.2. Mixing layer. Experimental setup and coordinate system.

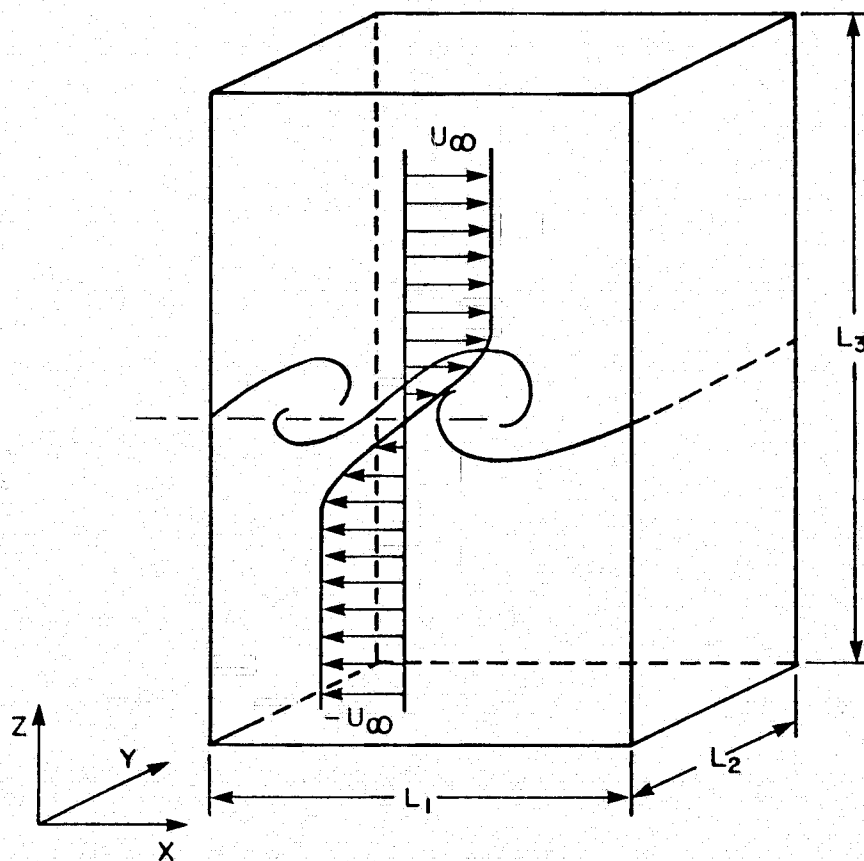


Fig. 5.3. Computational box and coordinate system.

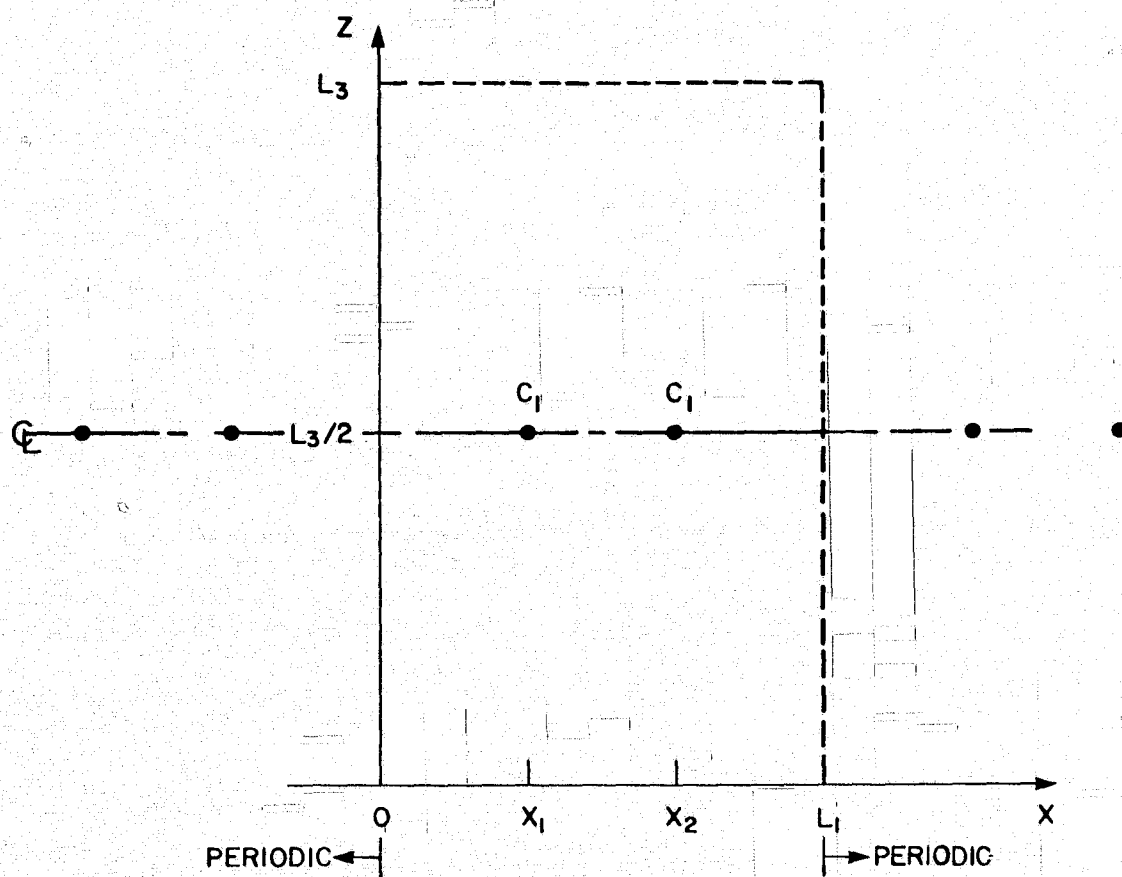


Fig. 5.4. Initial conditions setup $\left(\beta = \frac{1}{2} - \frac{|x_1 - x_2|}{L} \right)$

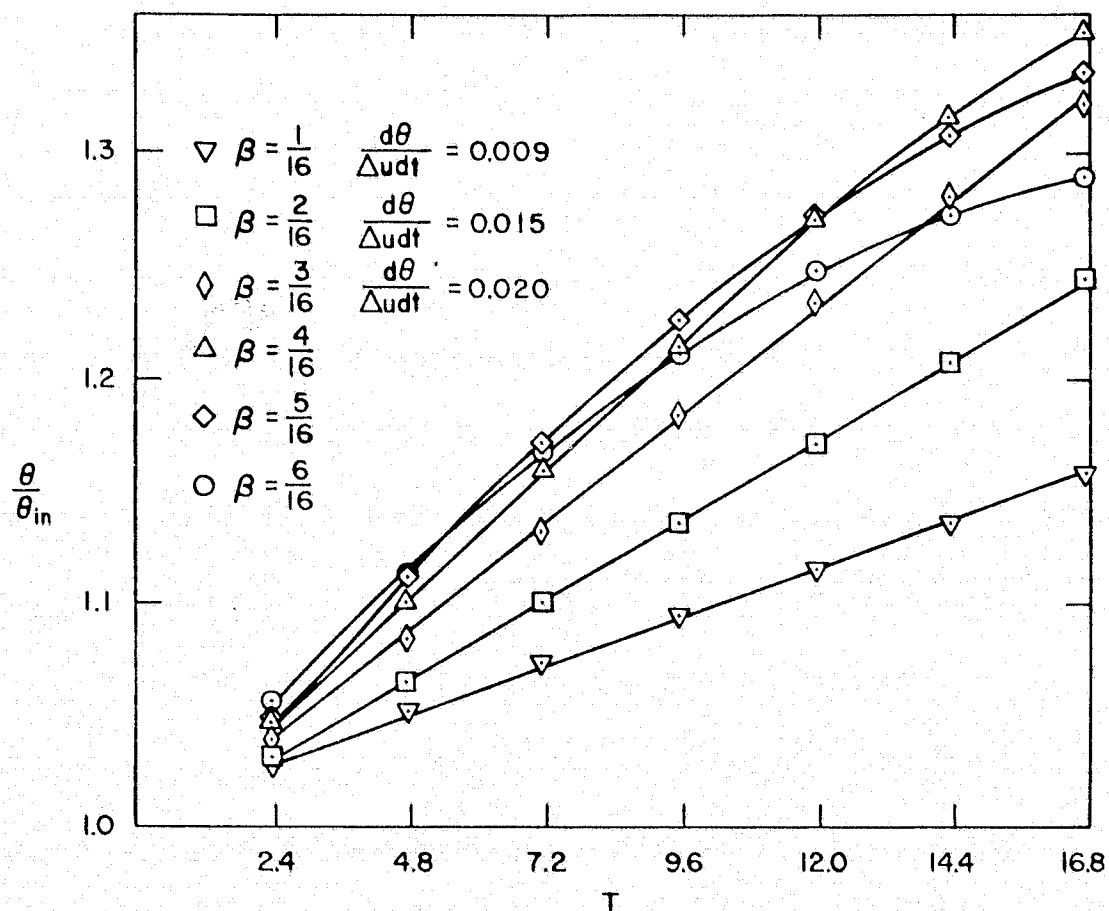


Fig. 5.5. Non-dimensional momentum thickness (θ/θ_{in}) as a function of time for various β . Two-dimensional computations.

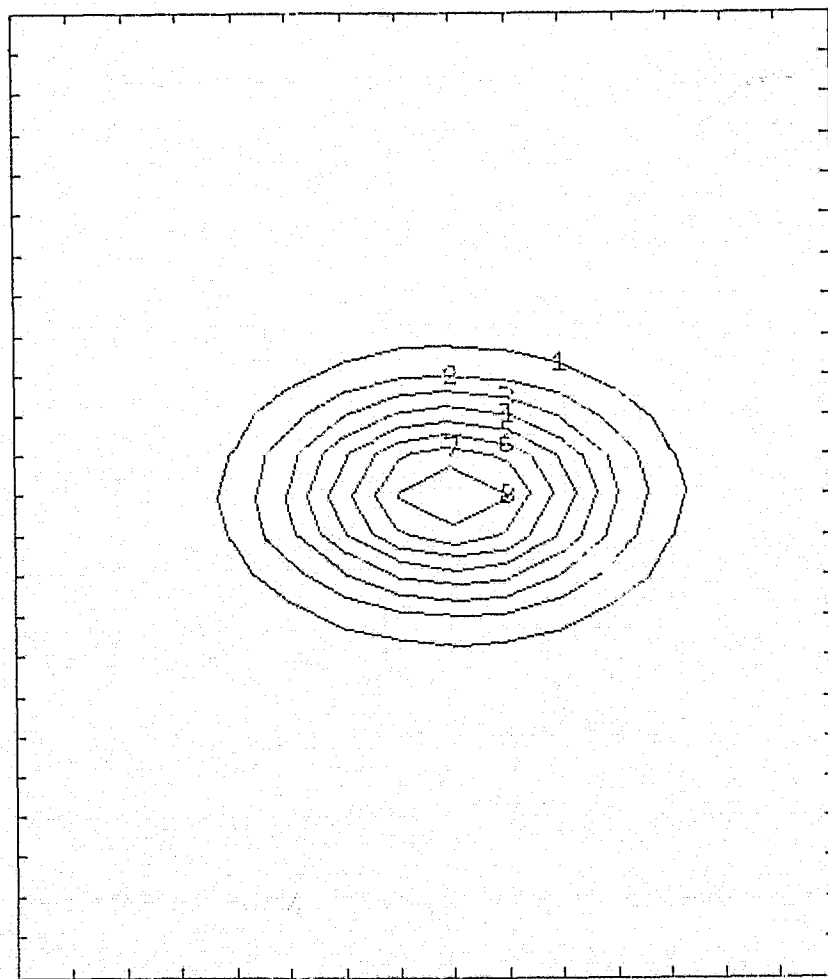


Fig. 5.6a. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 6/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\omega_{2,\max} = 0.702$).

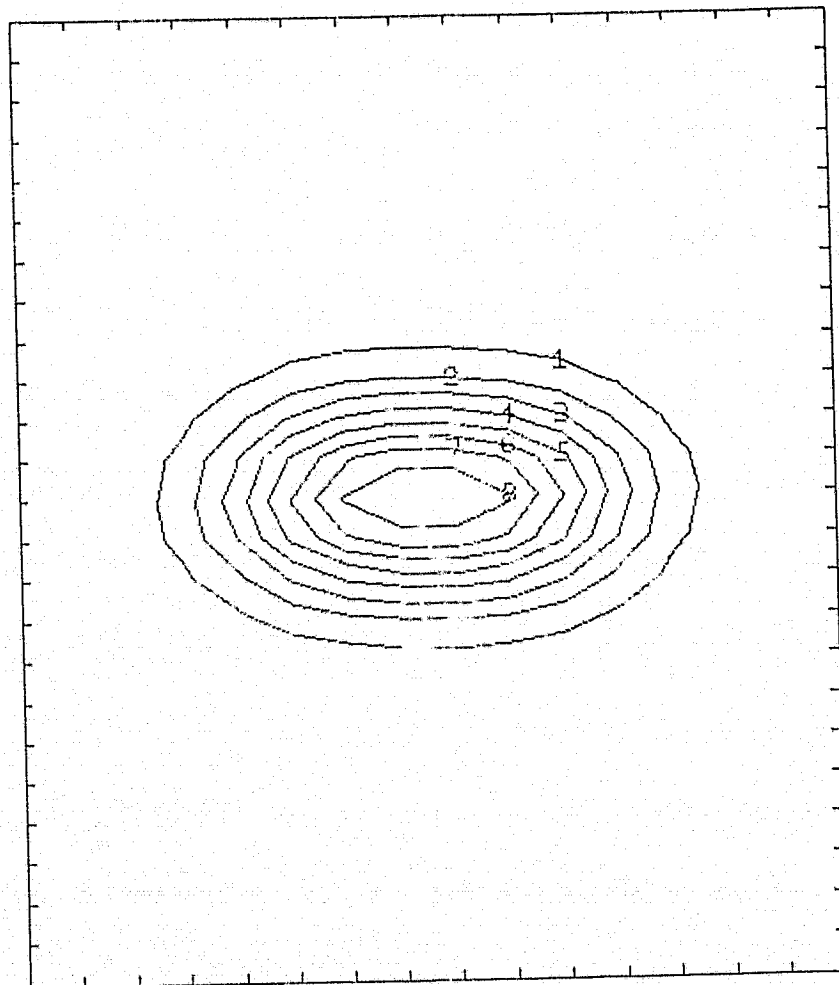


Fig. 5.6b. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 5/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.564$).

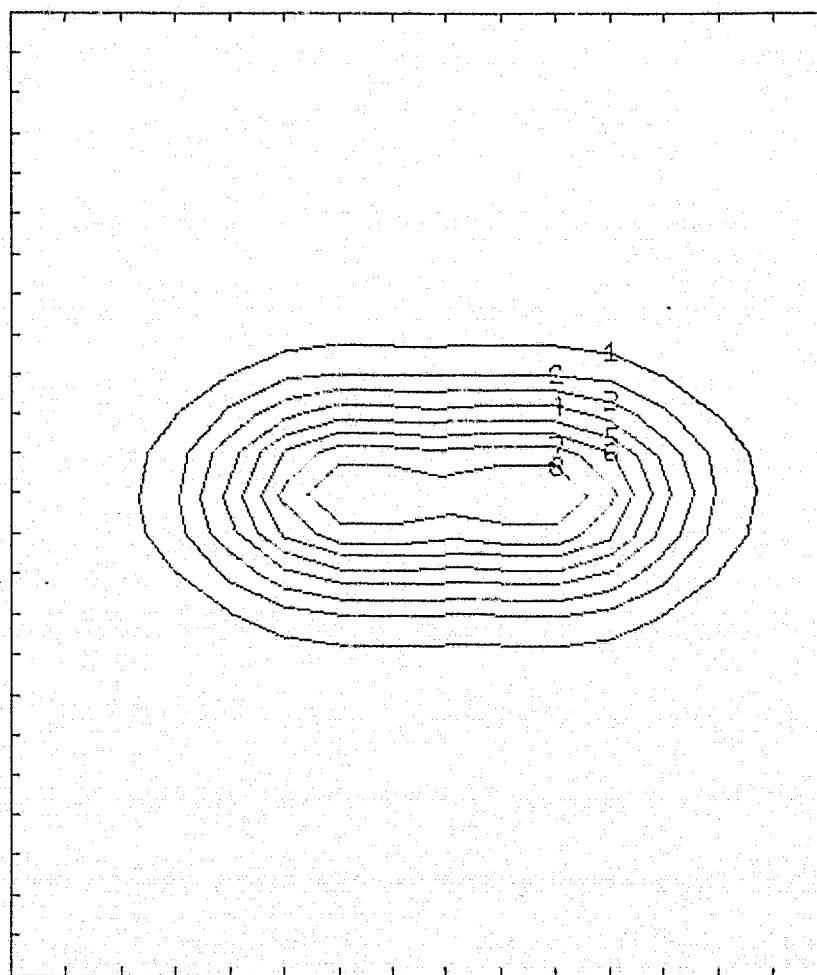


Fig. 5.6c. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 4/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\omega_{2,\max} = 0.444$).

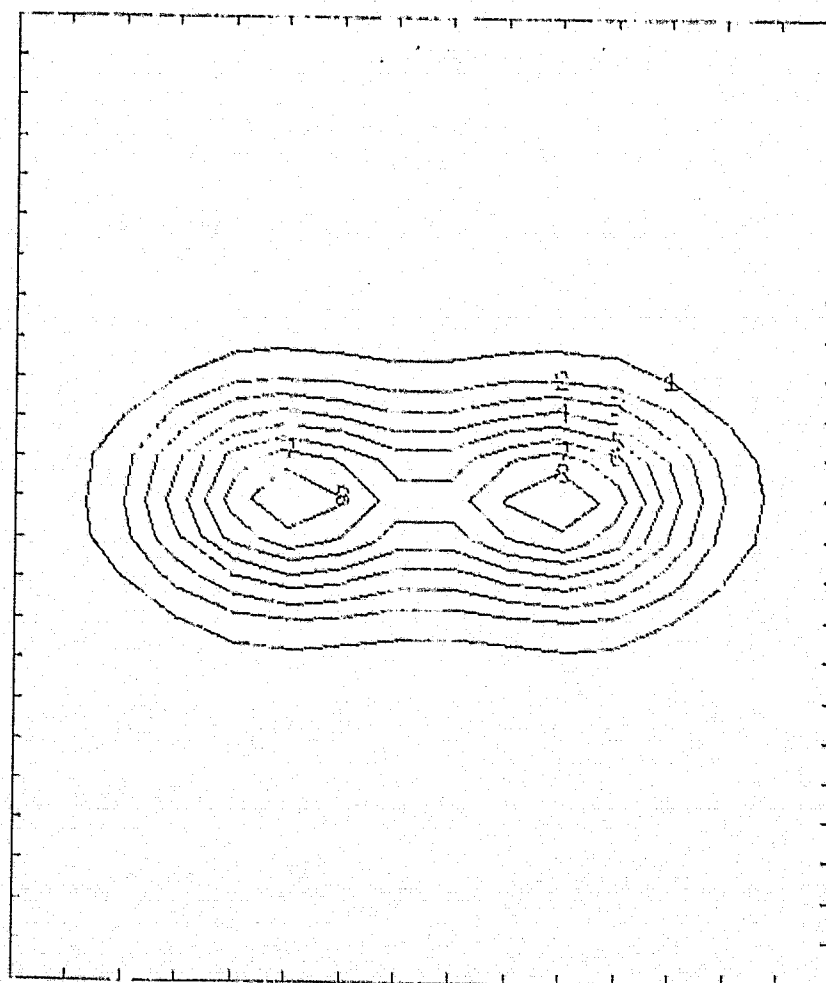


Fig. 5.6d. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 3/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\bar{\omega}_{2,\max} = 0.421$).

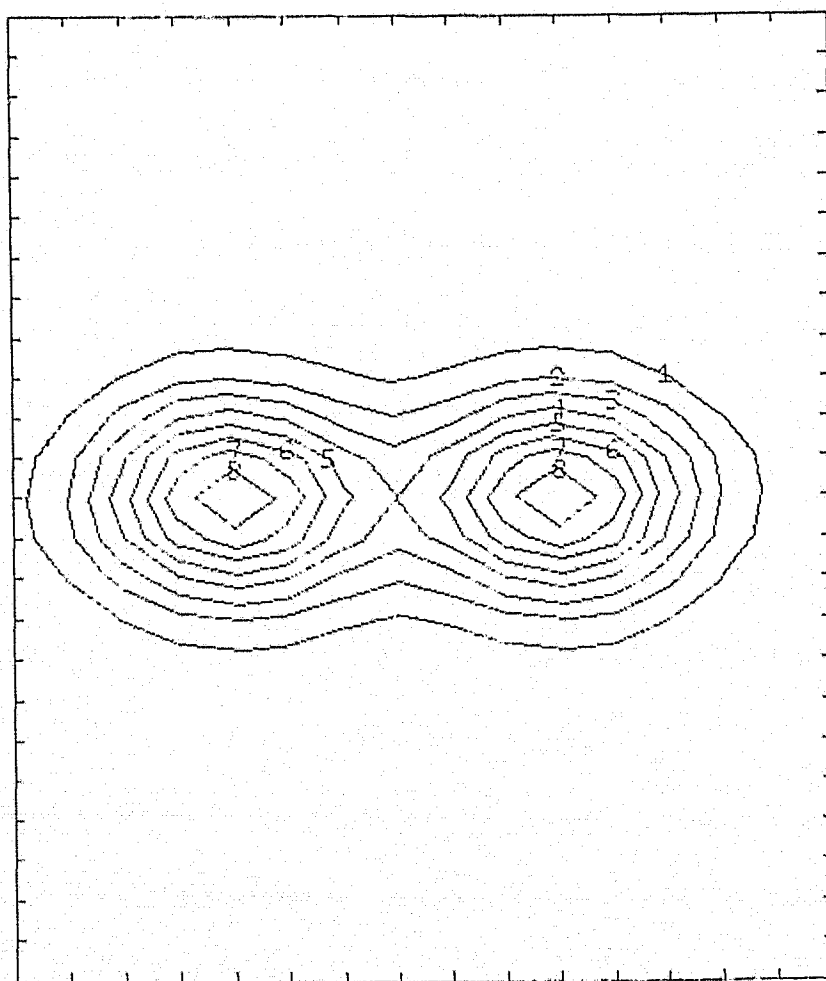


Fig. 5.6e. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 2/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\omega_{2,\max} = 0.416$).

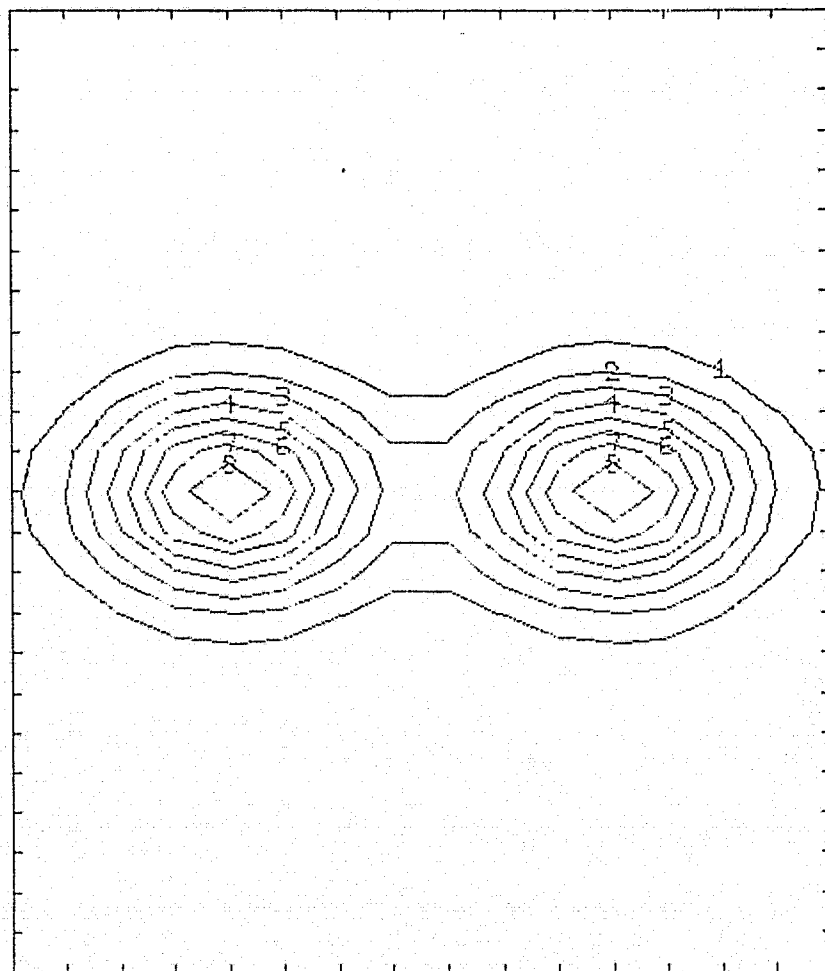


Fig. 5.6f. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 1/16$, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\omega_{2,\max} = 0.415$).

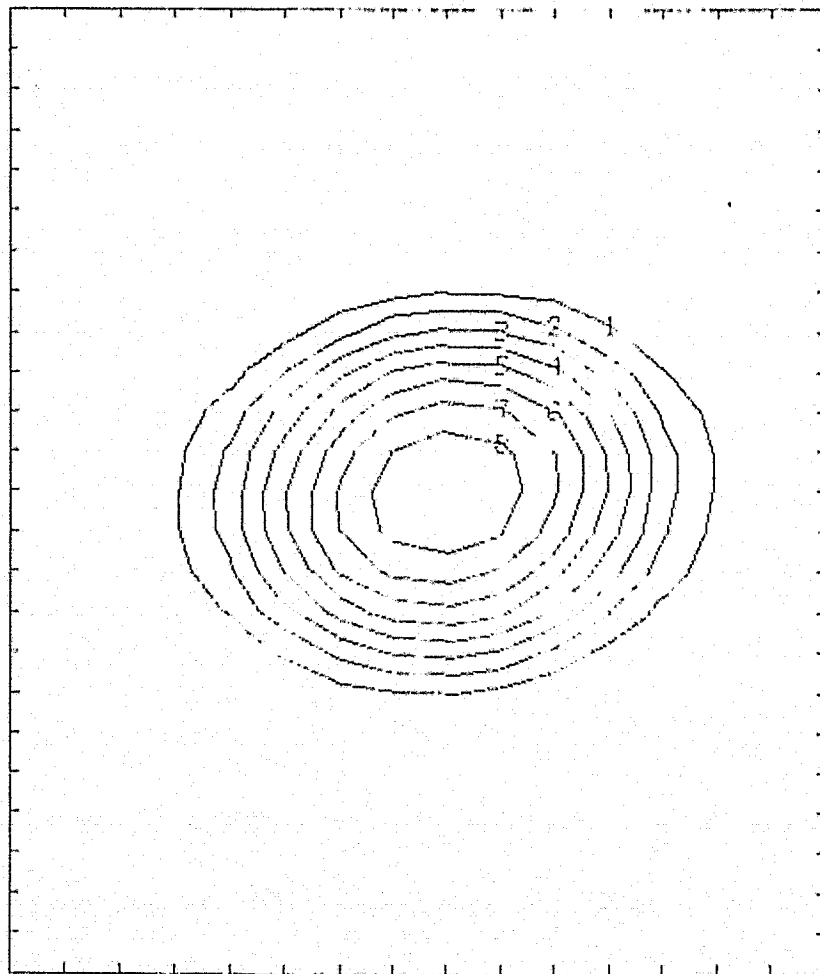


Fig. 5.7a. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 6/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.394$).

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

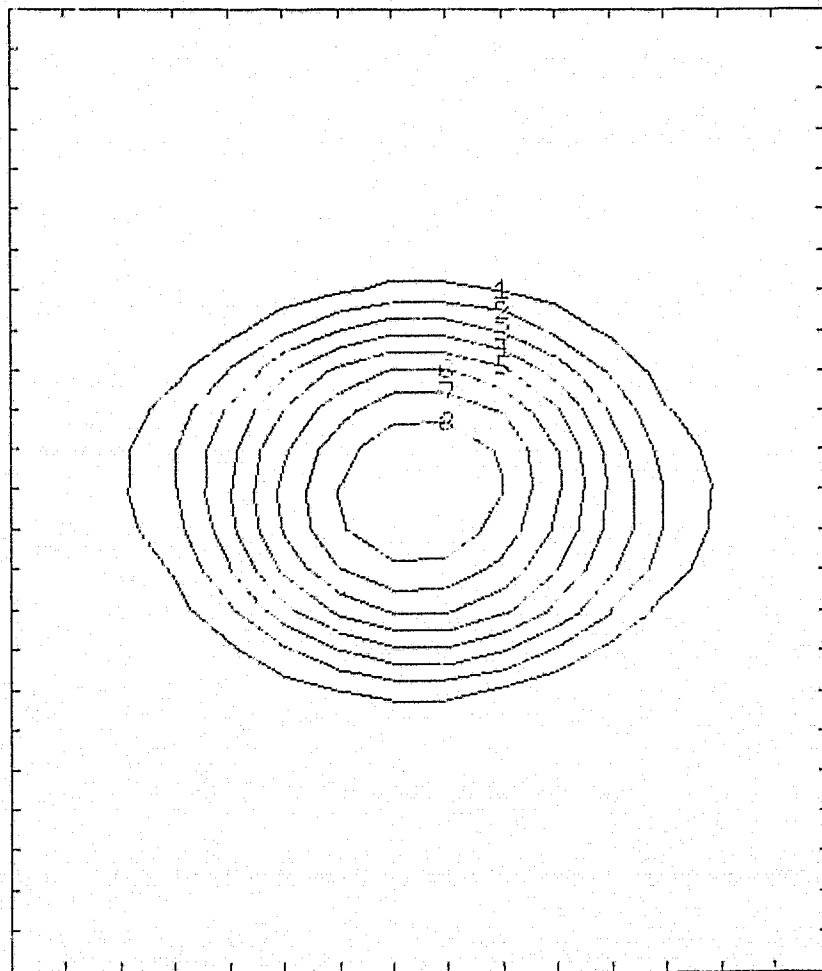


Fig. 5.7b. Contour plots of the spanwise vorticity $\bar{\omega}_2$ for $\beta = 5/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.358$).

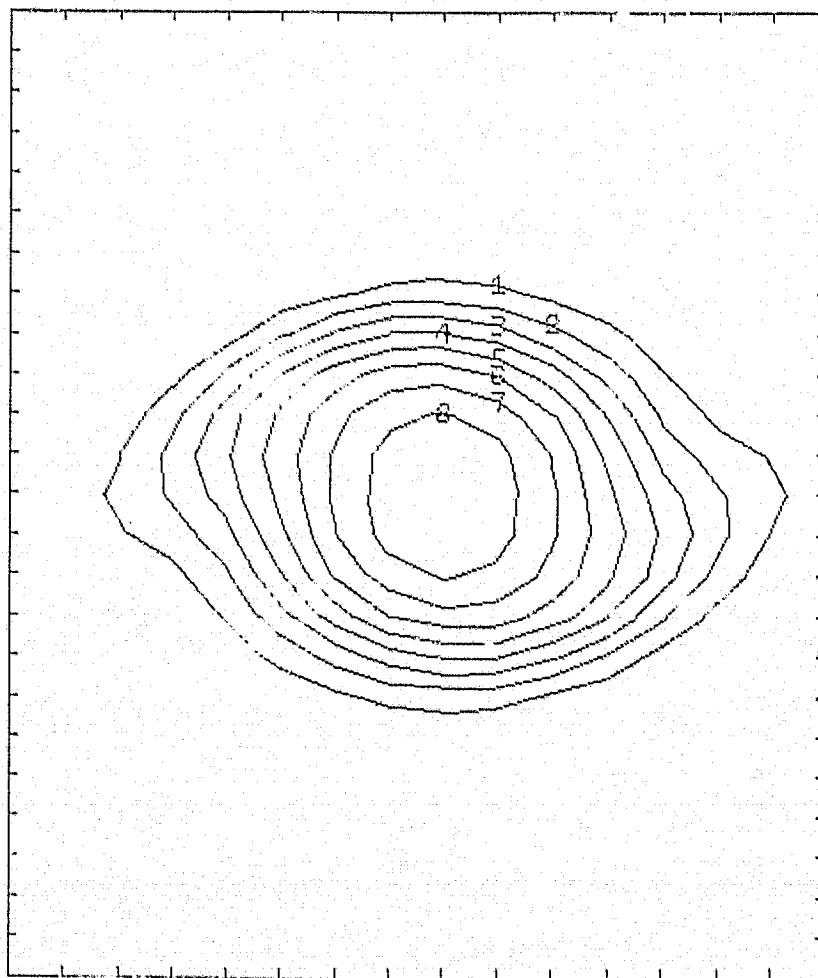


Fig. 5.7c. Contour plots of the spanwise vorticity (ω_2) for $\beta = 4/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\omega_{2,\max} = 0.322$).

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

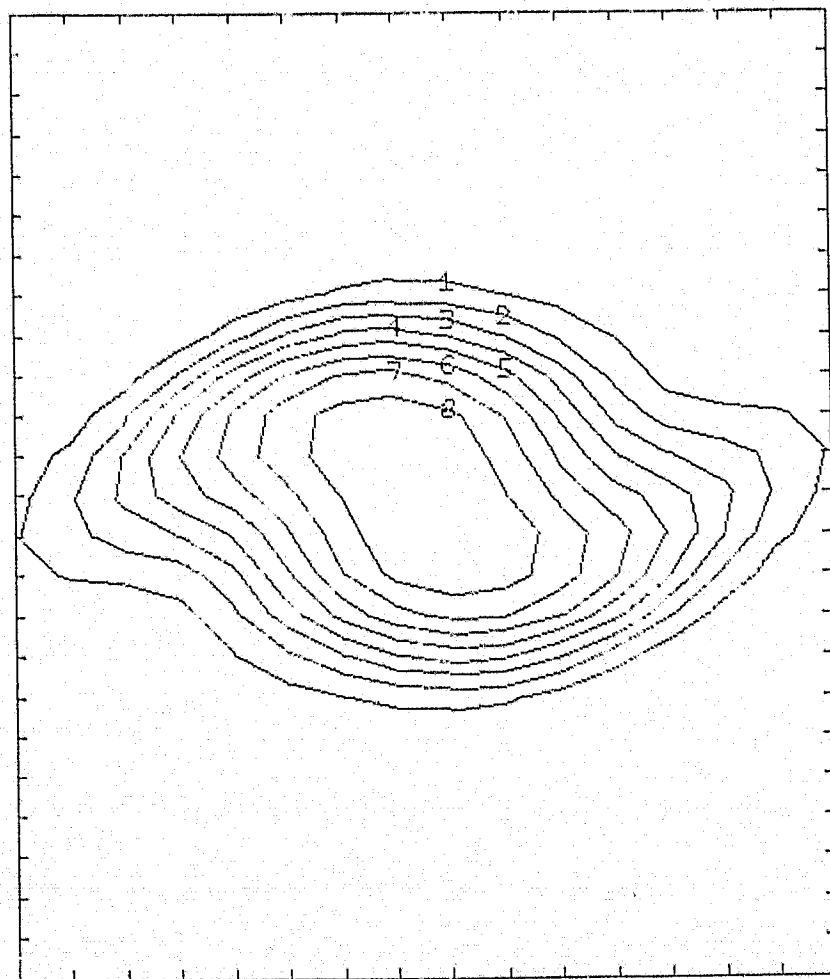


Fig. 5.7d. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for $\beta = 3/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\bar{\omega}_{2,\max} = 0.276$).

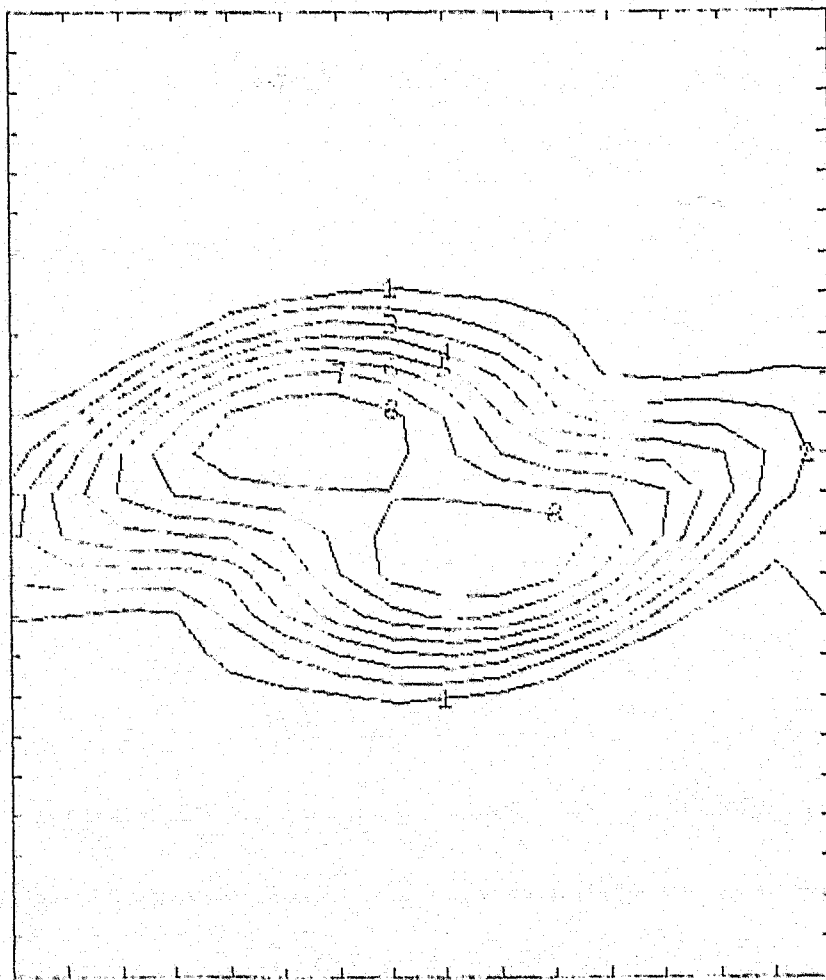


Fig. 5.7e. Contour plots of the spanwise vorticity (ω_2) for $\beta = 2/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\omega_{2,\max} = 0.248$).

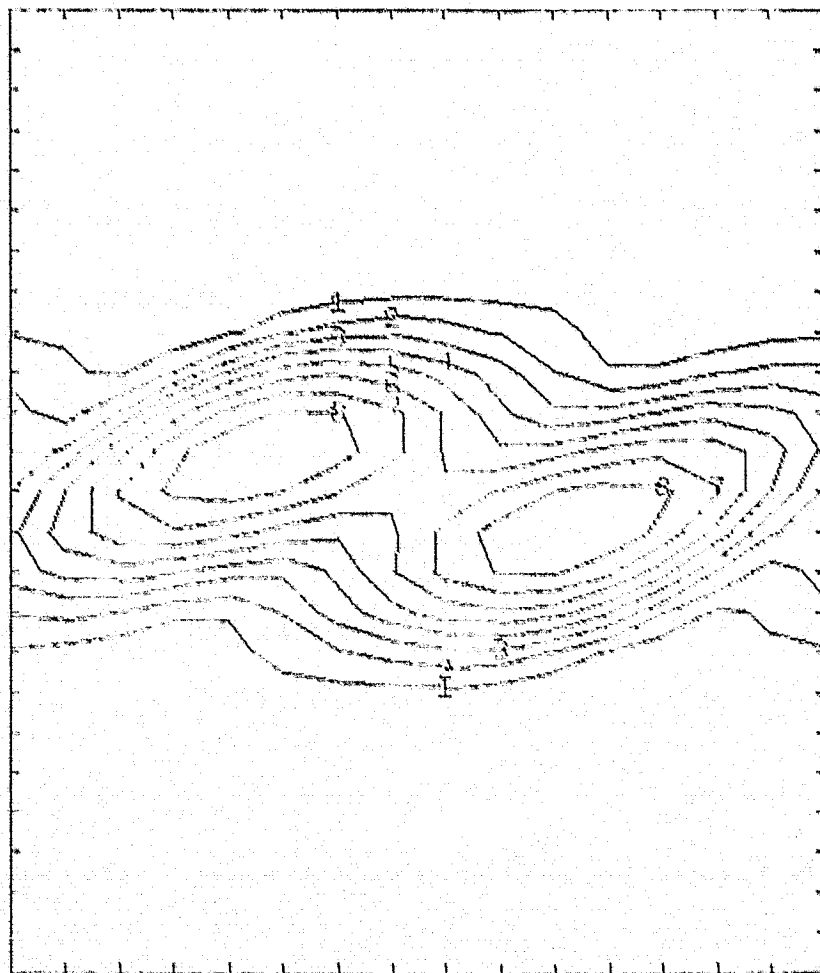


Fig. 5.7f. Contour plots of the spanwise vorticity ($\overline{w_2}$) for $\beta = 1/16$, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels ($\overline{w_{2,\max}} = 0.245$).

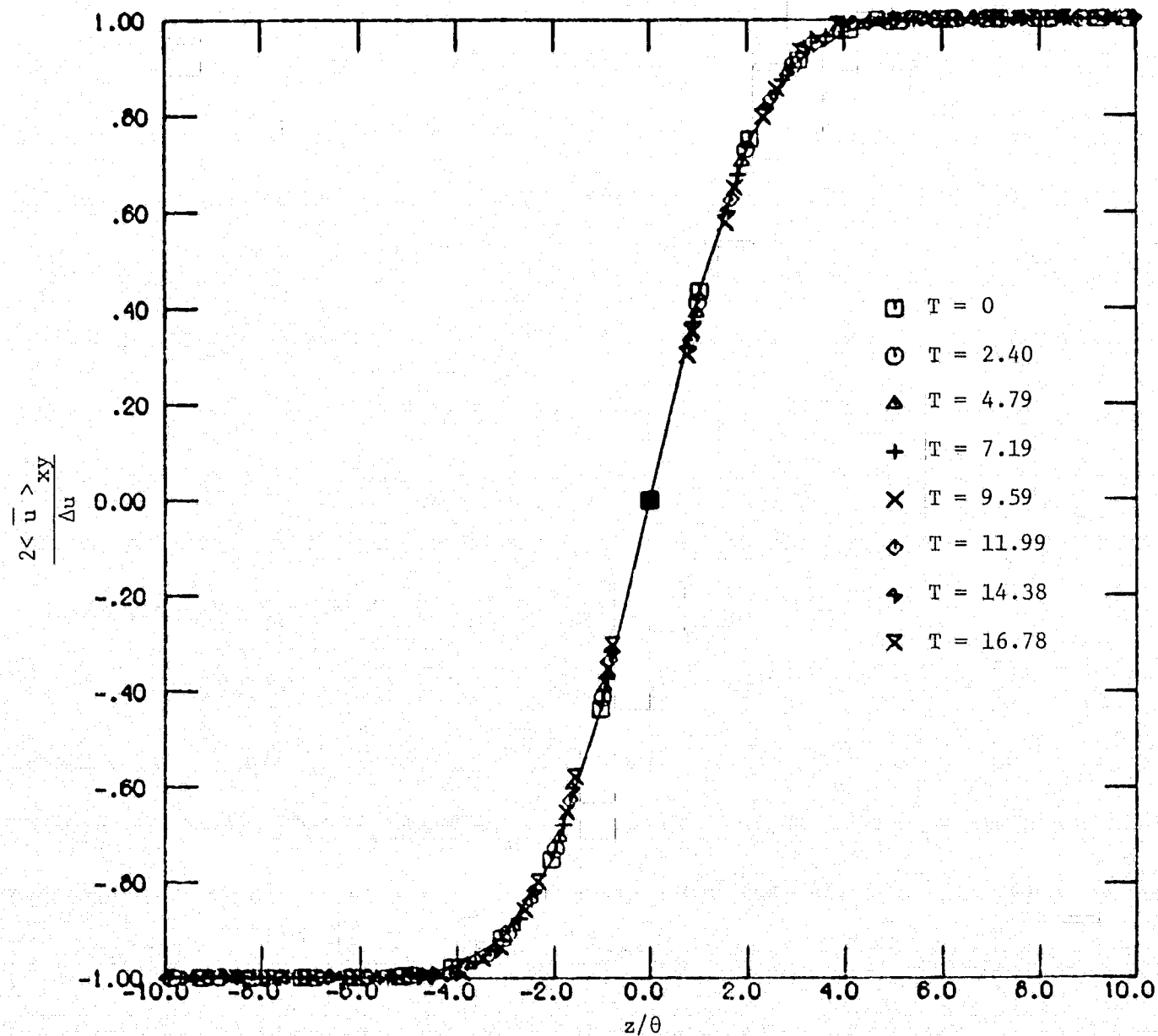


Fig. 5.8. Mean velocity profiles. Two-dimensional computations; ($\beta = 3/16$).

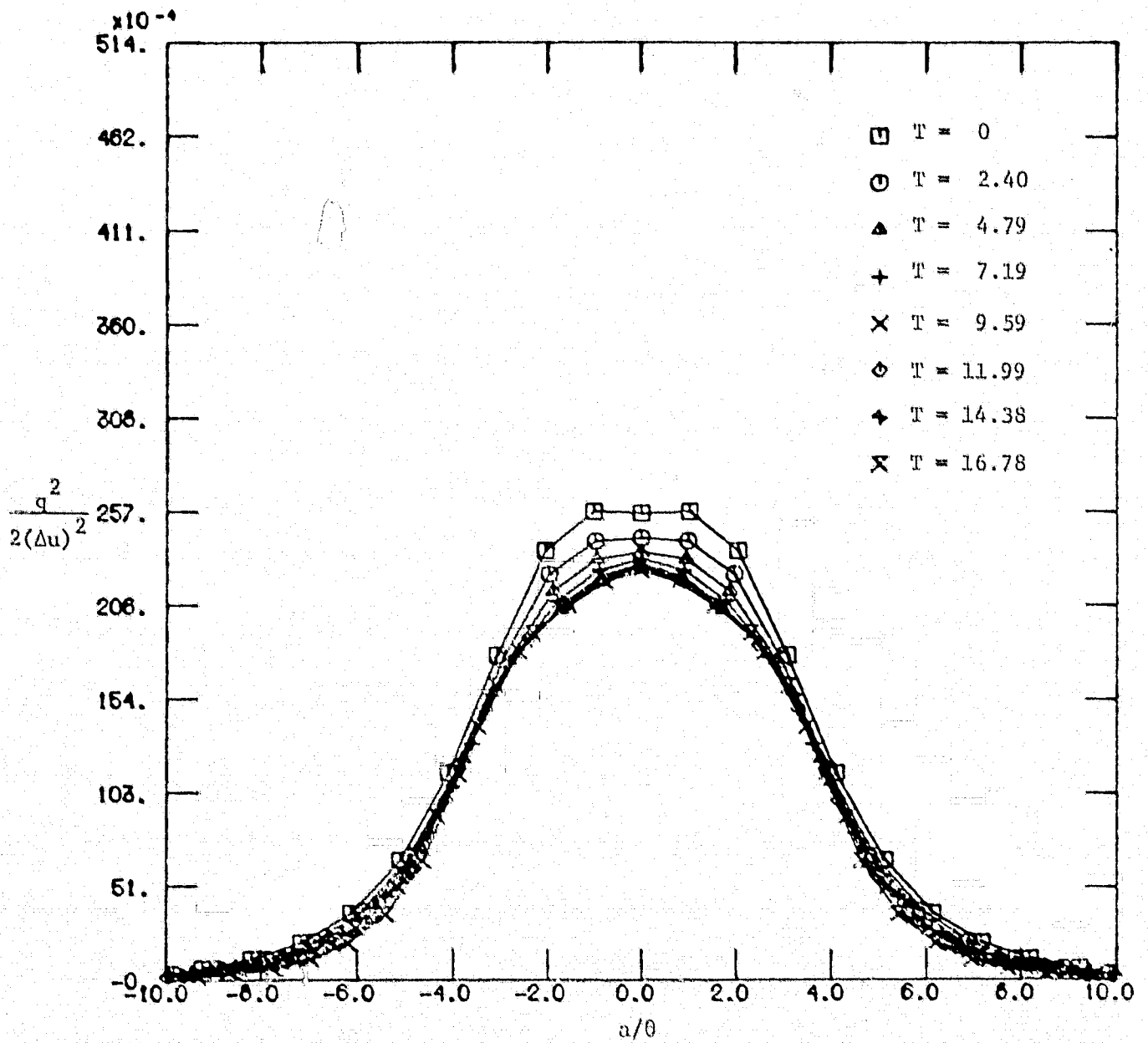


Fig. 5.9. Mean turbulence intensity profiles. Two-dimensional computations ($\beta = 3/16$).

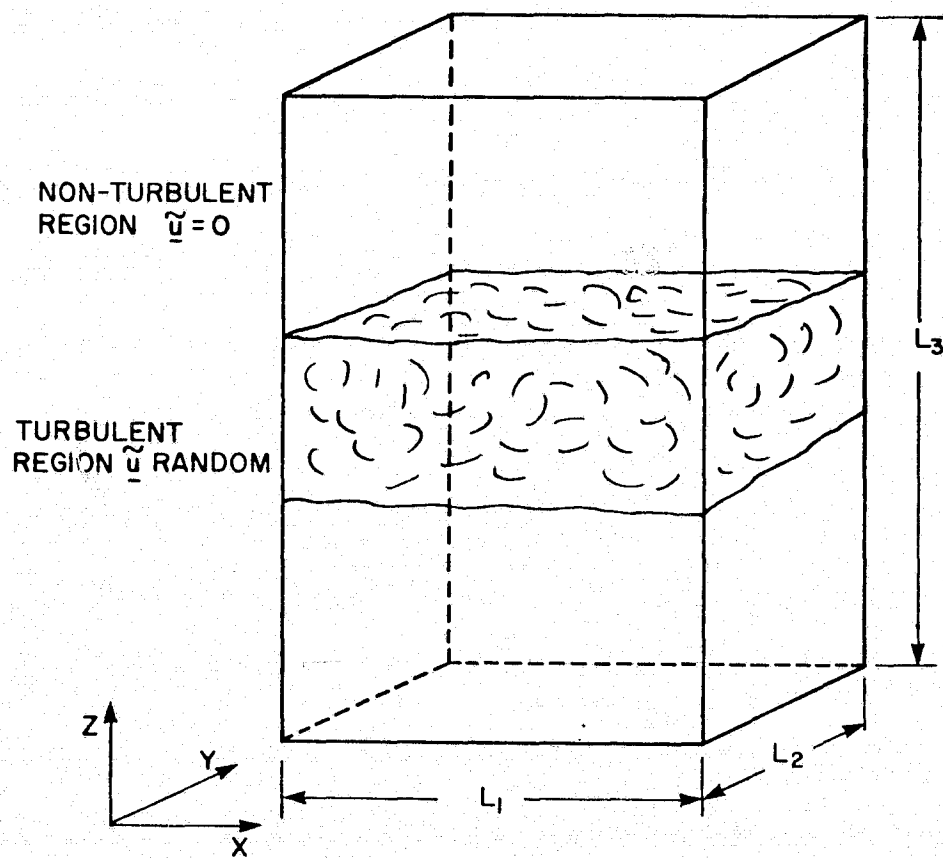
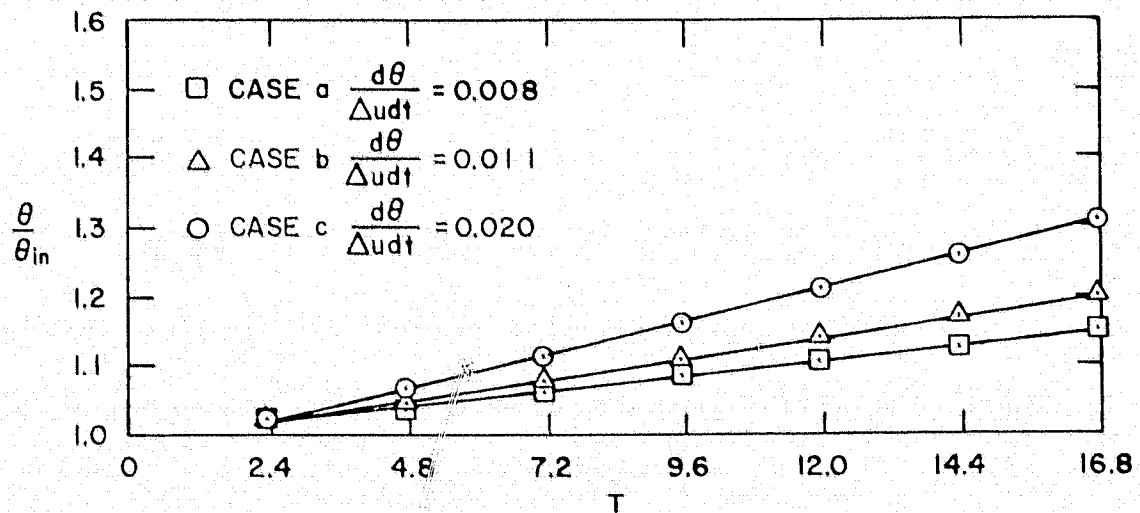


Fig. 6.1. Three-dimensional computation box. Random velocity setup and coordinate system.

Table 6.1

Three-Dimensional Computations of Turbulent Mixing Layers

Case	Amplitude of Random Field	Initial Conditions
a	$\frac{ u_i _{\max}}{\Delta u} = 0.01$	Random field + mean
b	$\frac{ u_i _{\max}}{\Delta u} = 0.30$	Random field + mean
c	$\frac{ u_i _{\max}}{\Delta u} = 0.30$	Random field + 2 spanwise vortices ($\beta = 3/16$)

Fig. 6.2. Non-dimensional momentum thickness (θ/θ_{in}) as a function of time. Three-dimensional computations.

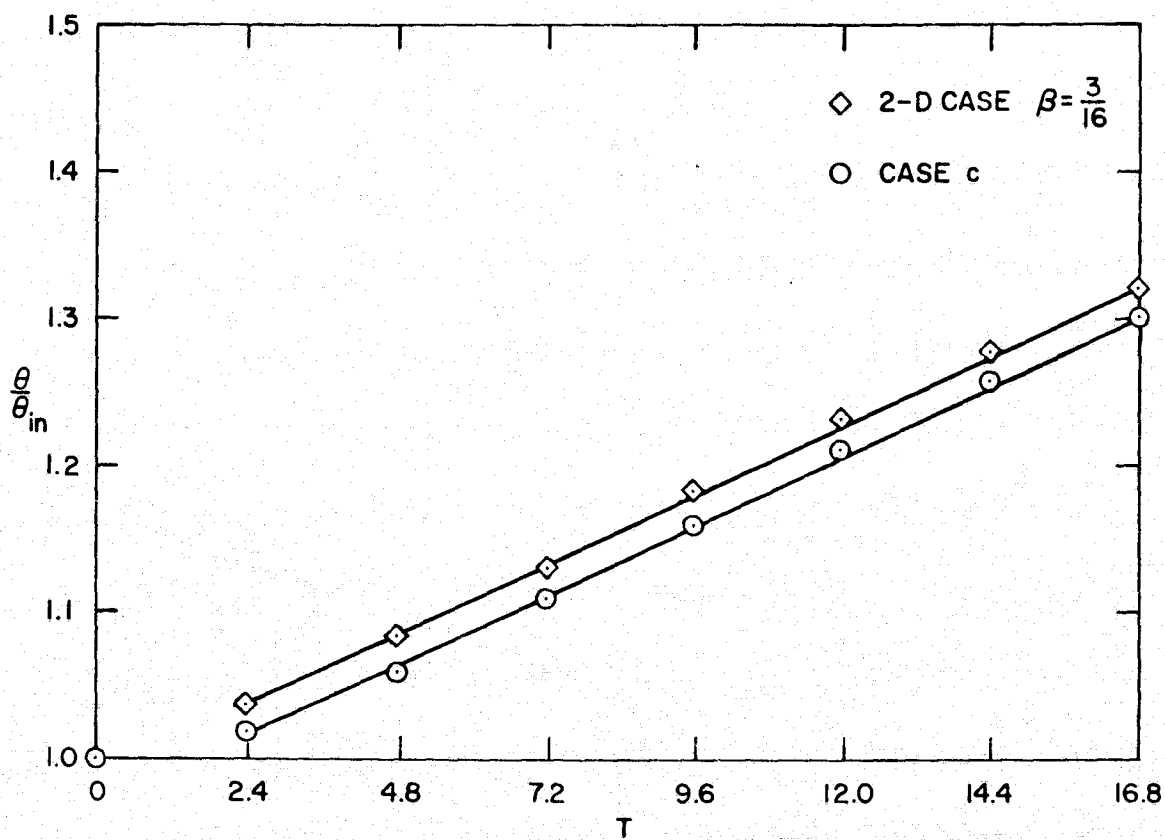


Fig. 6.3. Non-dimensional momentum thickness (θ/θ_{in}) as a function of time.

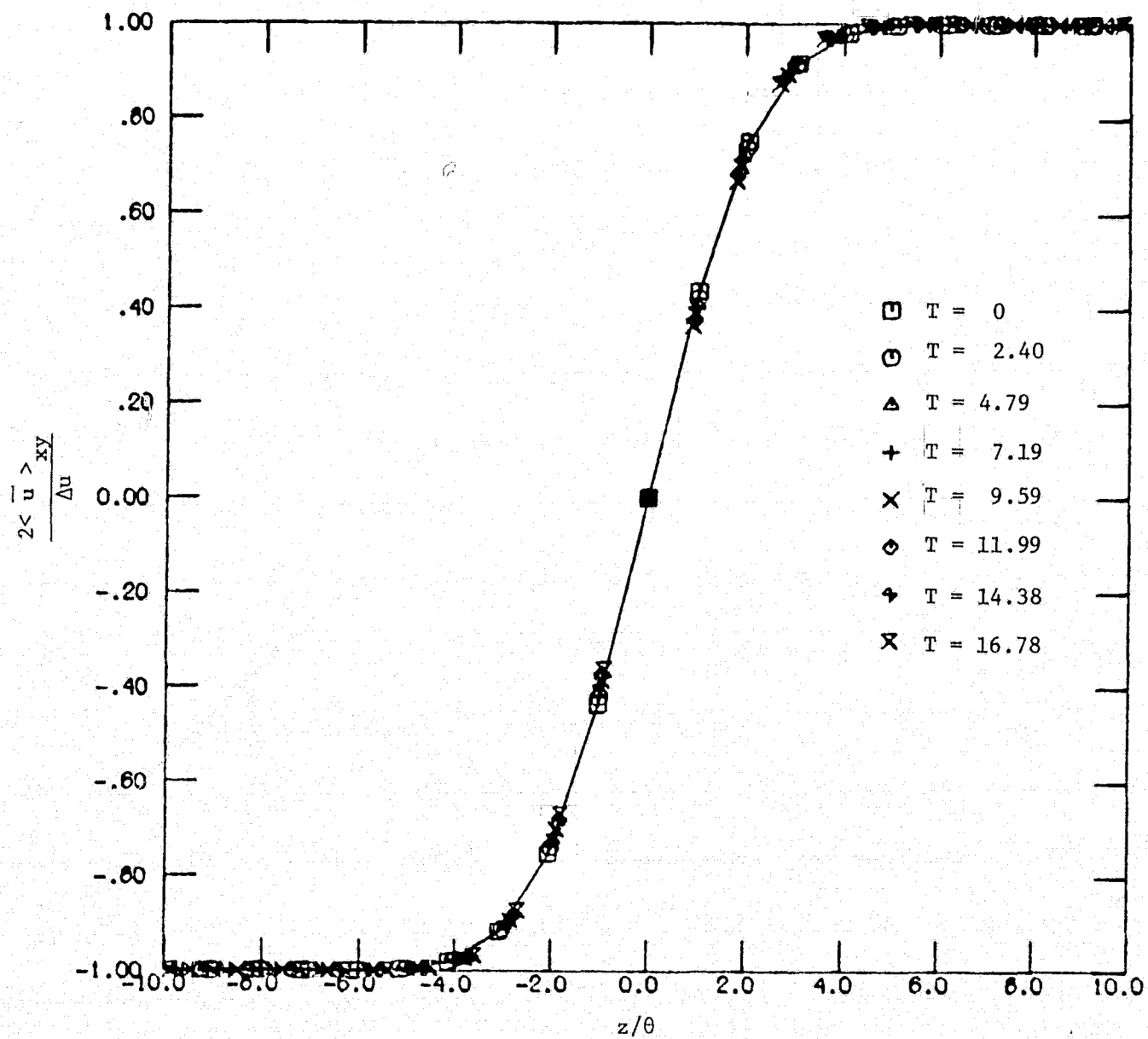


Fig. 6.4a. Mean velocity profiles. Three-dimensional computation (case a).

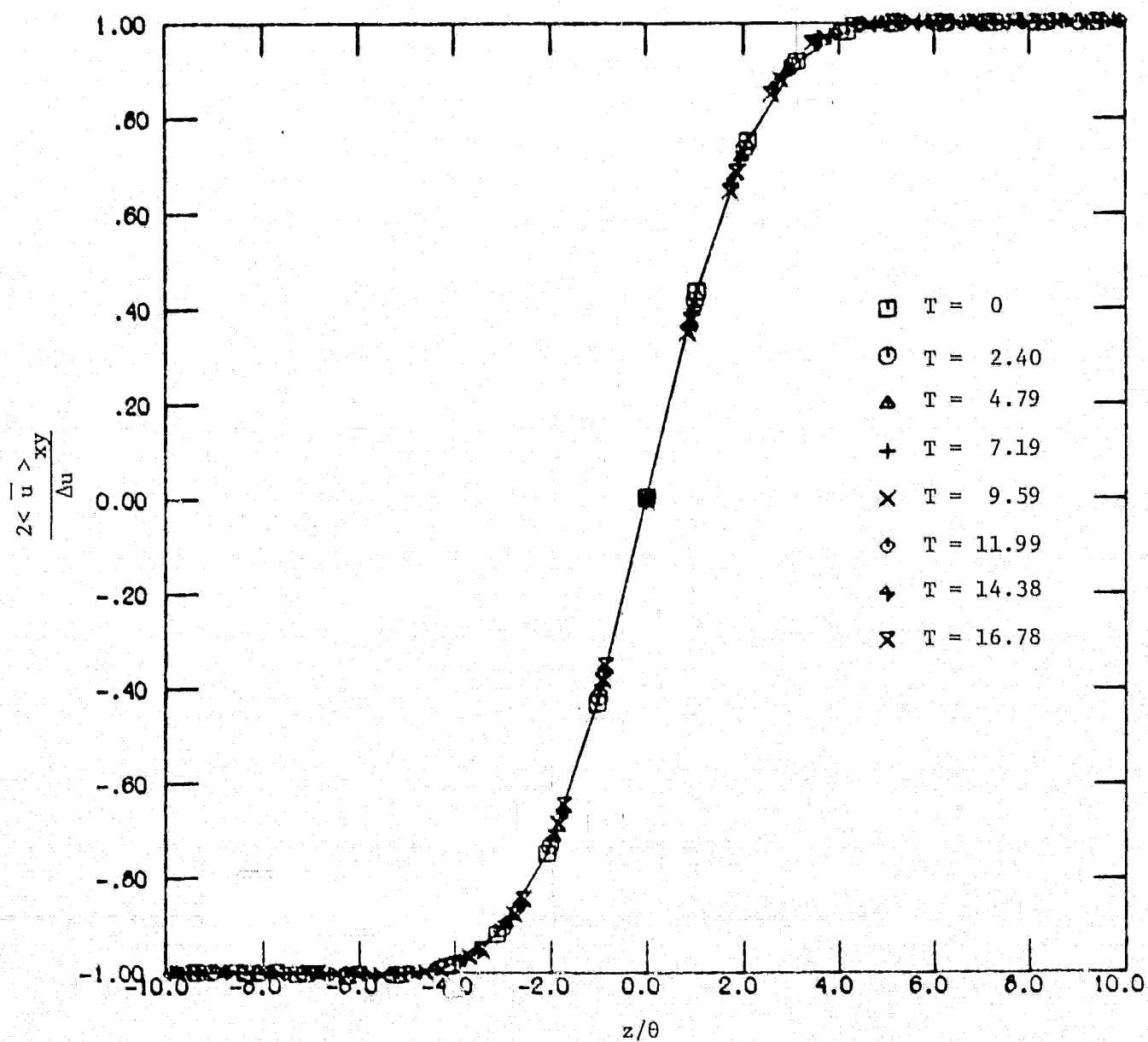


Fig. 6.4b. Mean velocity profiles. Three-dimensional computation (case b).

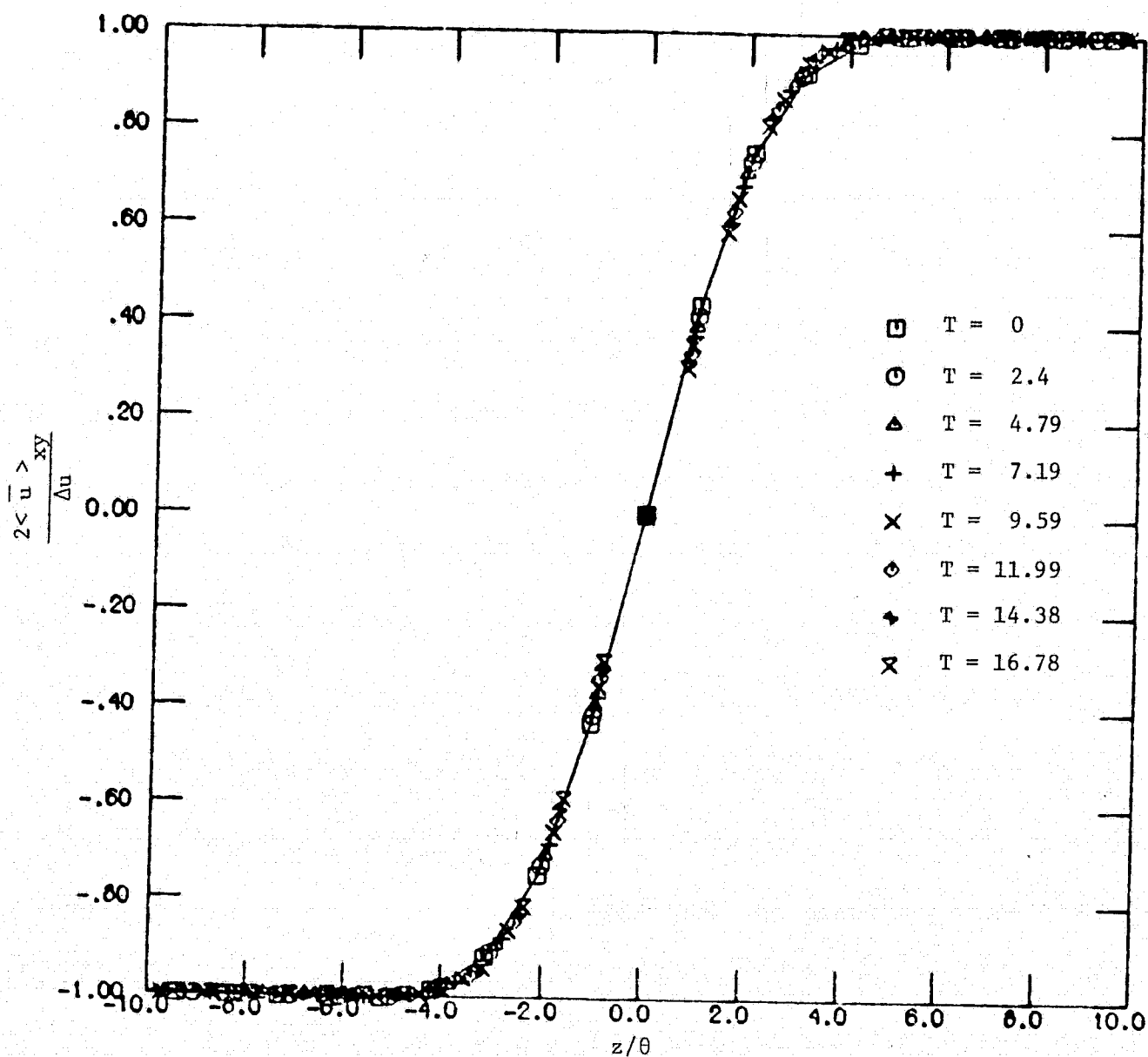


Fig. 6.4c. Mean velocity profiles. Three-dimensional computations (case c).

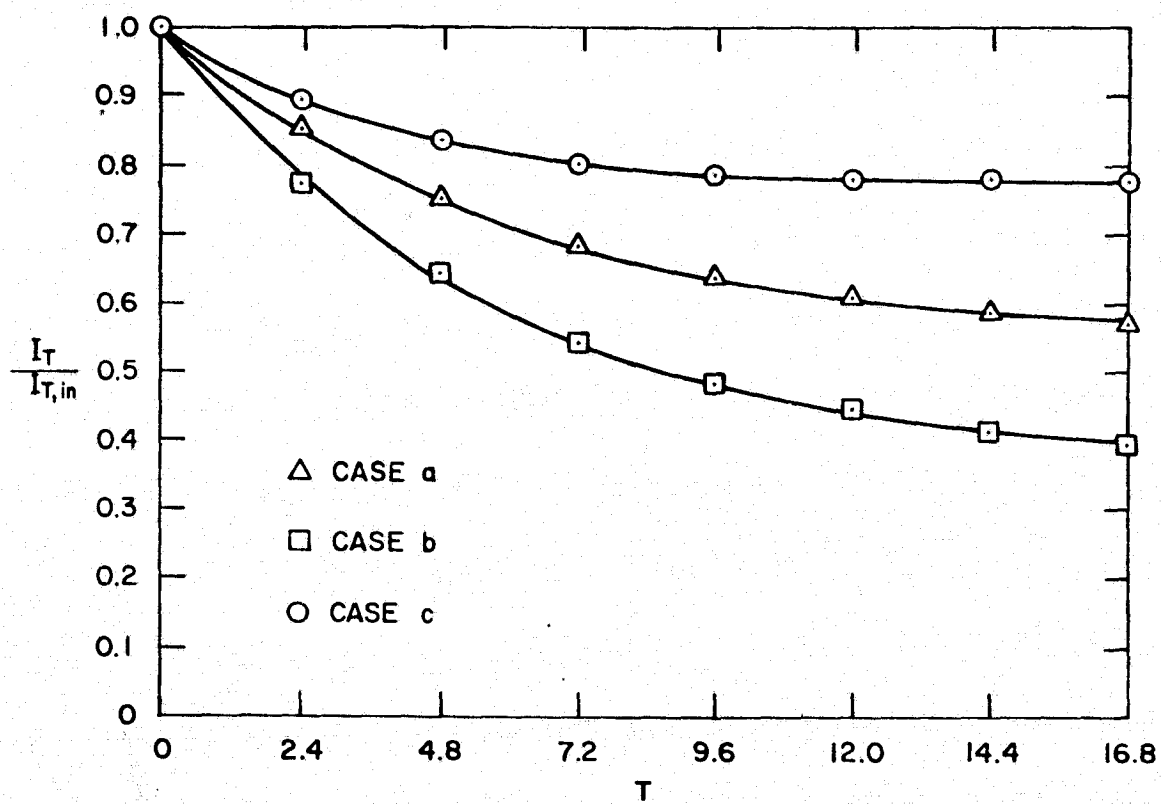


Fig. 6.5. Integral of the turbulence energy as a function of time.

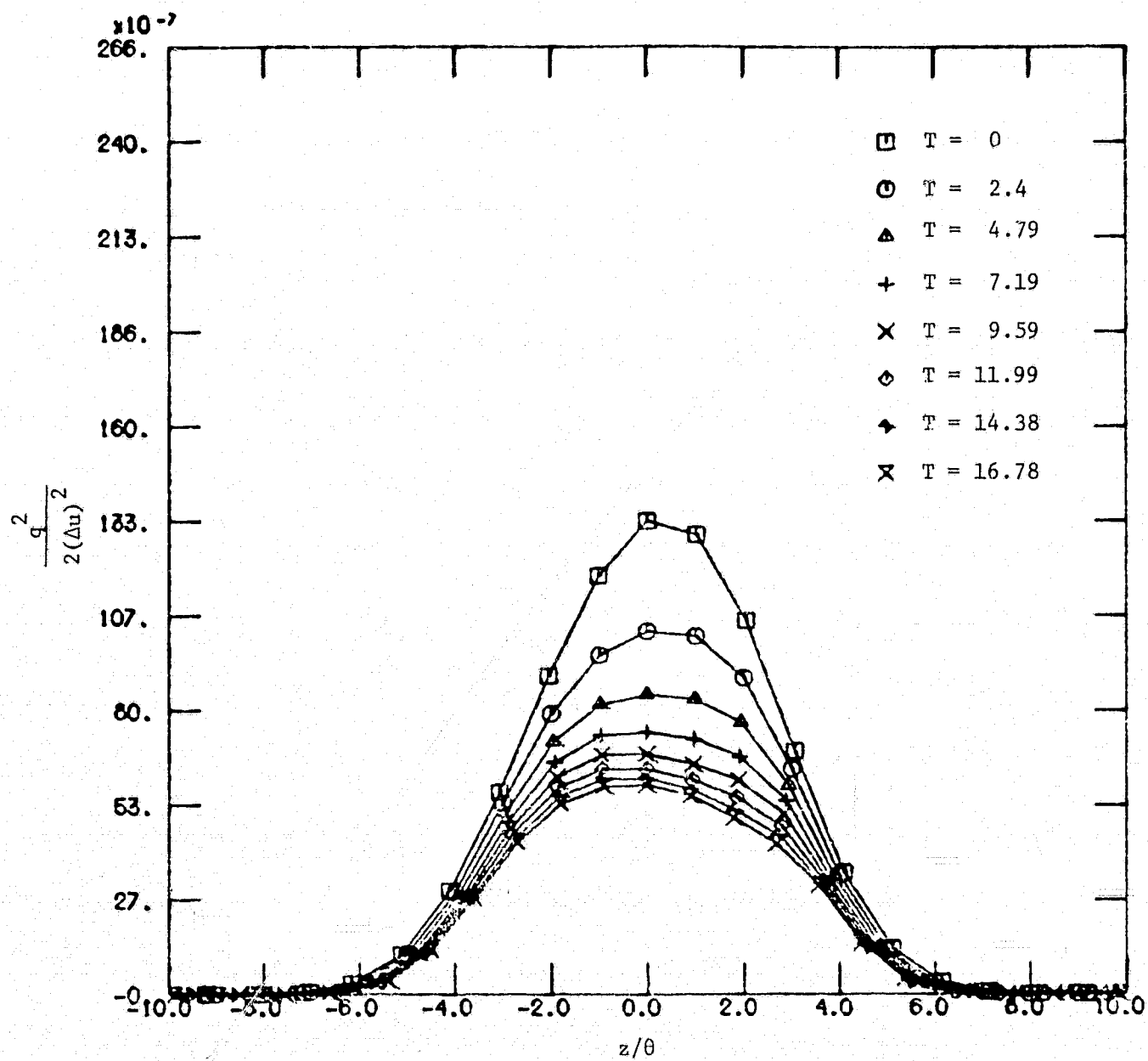


Fig. 6.6a. Mean turbulence intensity profiles. Three-dimensional computations (case a).

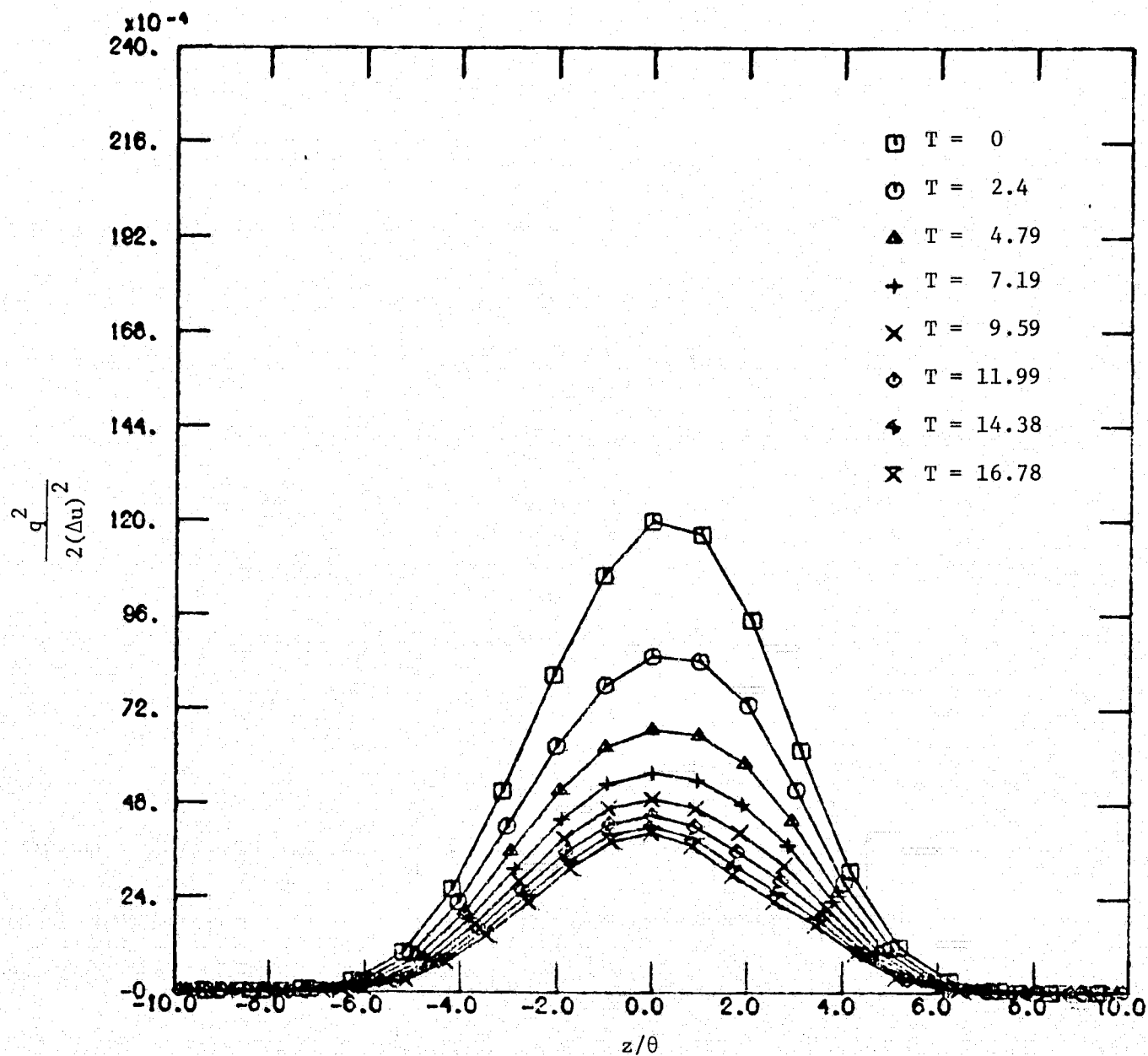


Fig. 6.6b. Mean turbulence intensity profiles. Three-dimensional computations (case b).

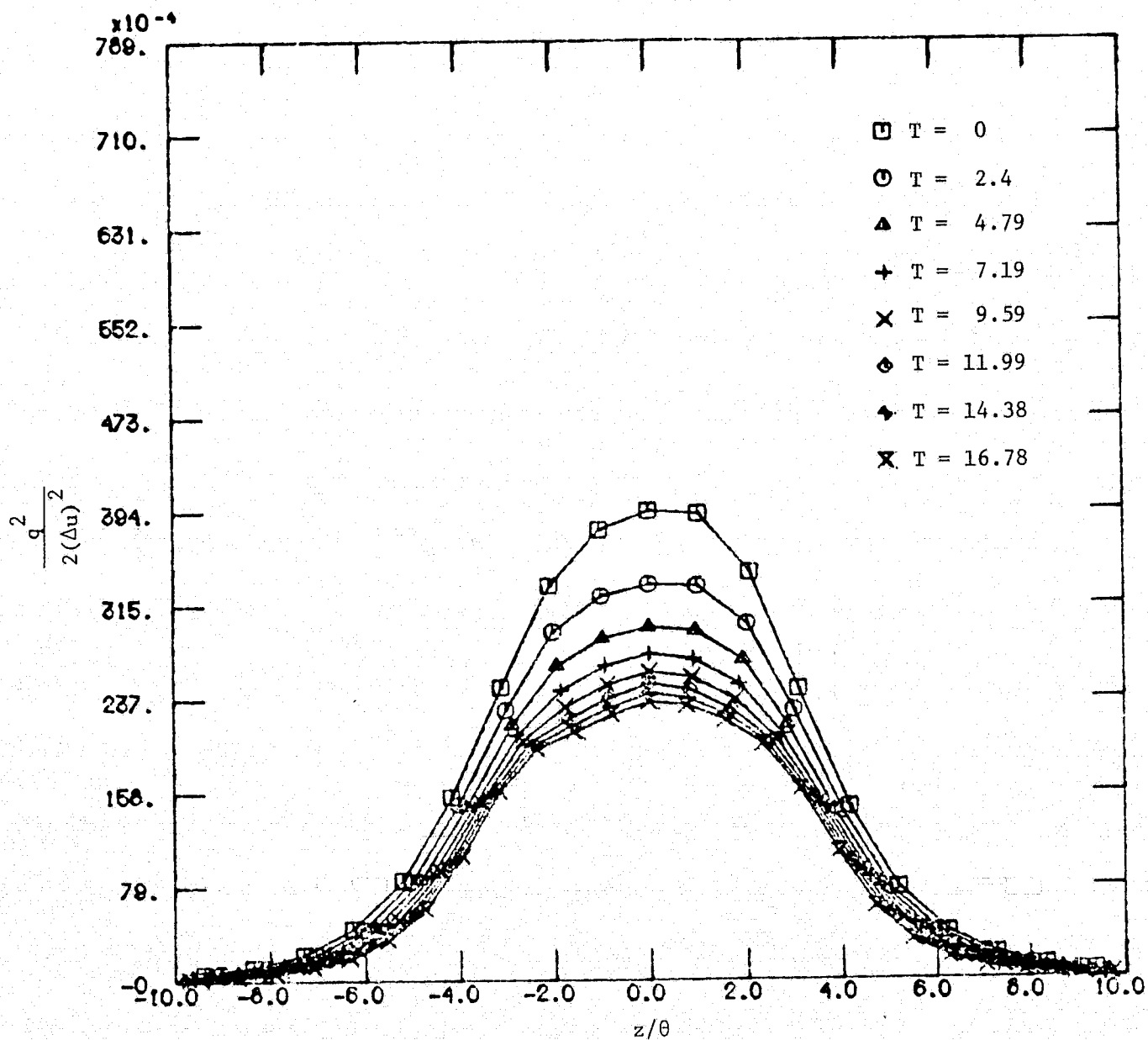


Fig. 6.6c. Mean turbulence intensity profiles. Three-dimensional computations (case c).

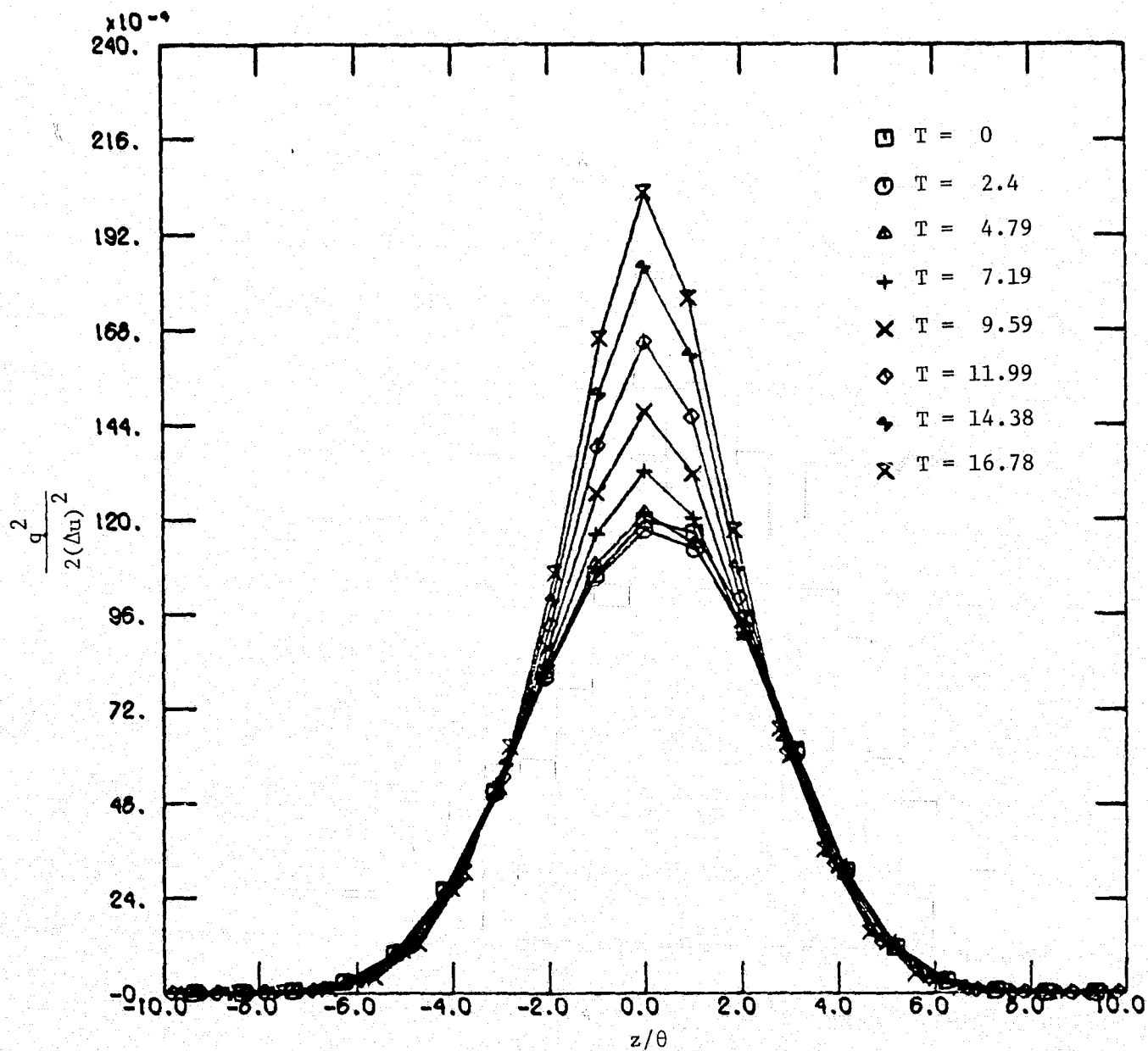


Fig. 6.7. Mean turbulence intensity profiles. Three-dimensional computation ($C_v = 0$).

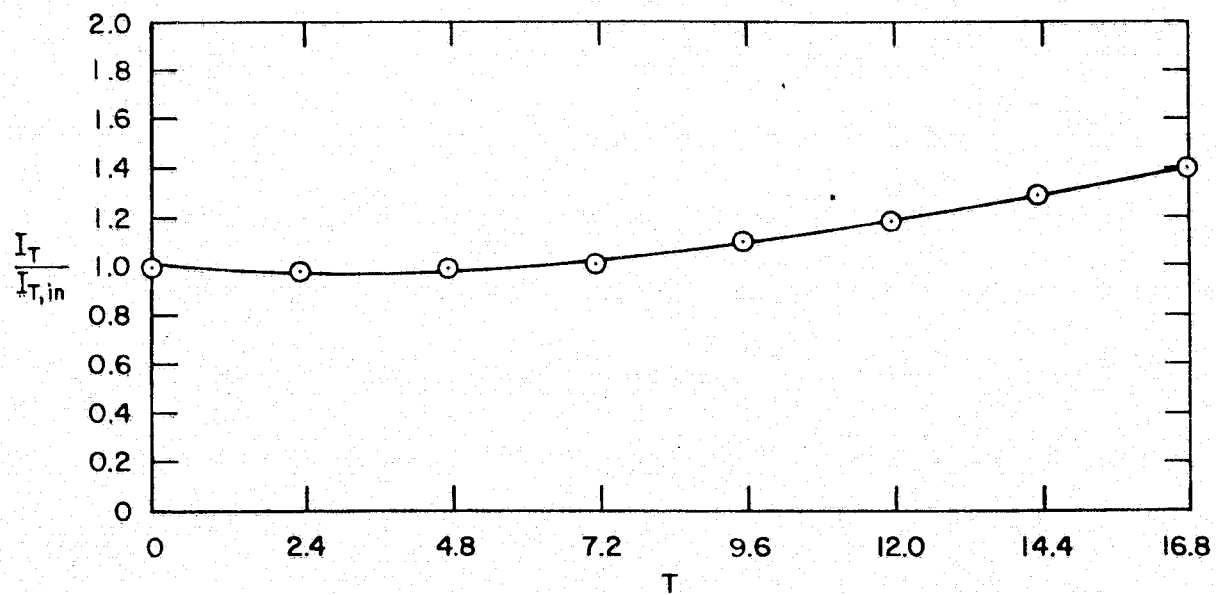


Fig. 6.8. Integral of the turbulence energy as a function of time. Three-dimensional computation ($C_v = 0$).

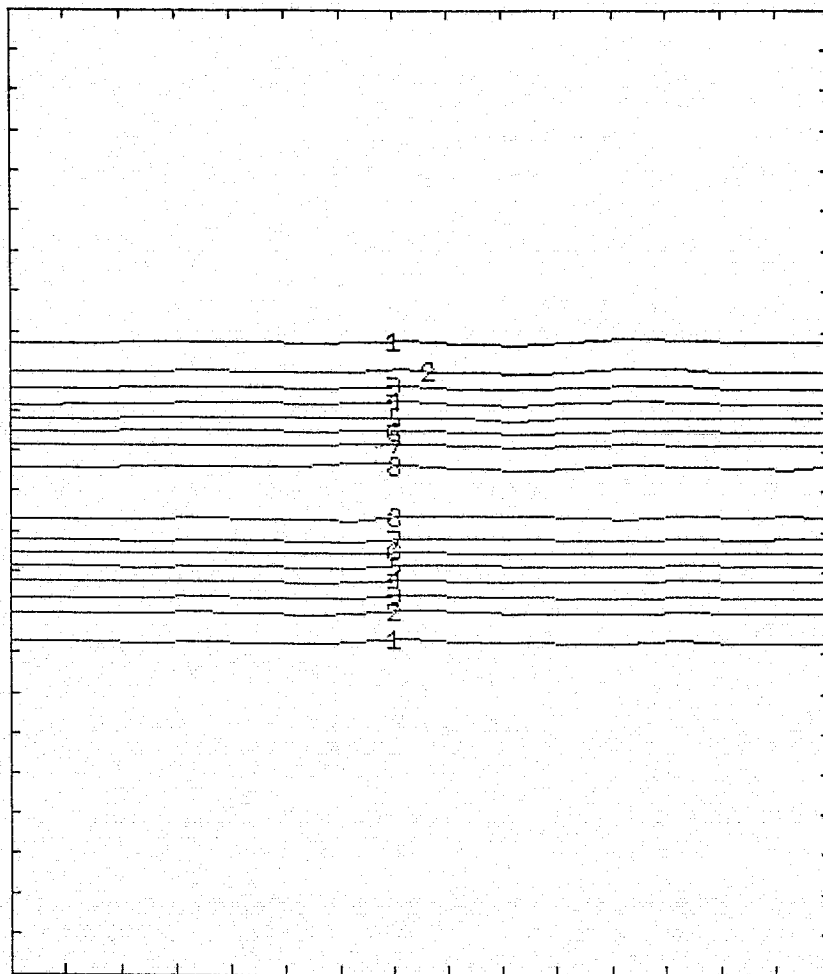
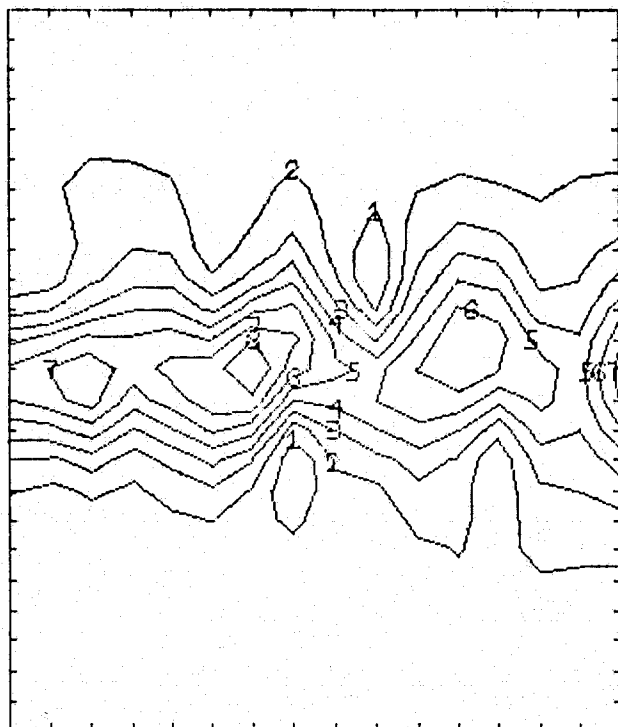
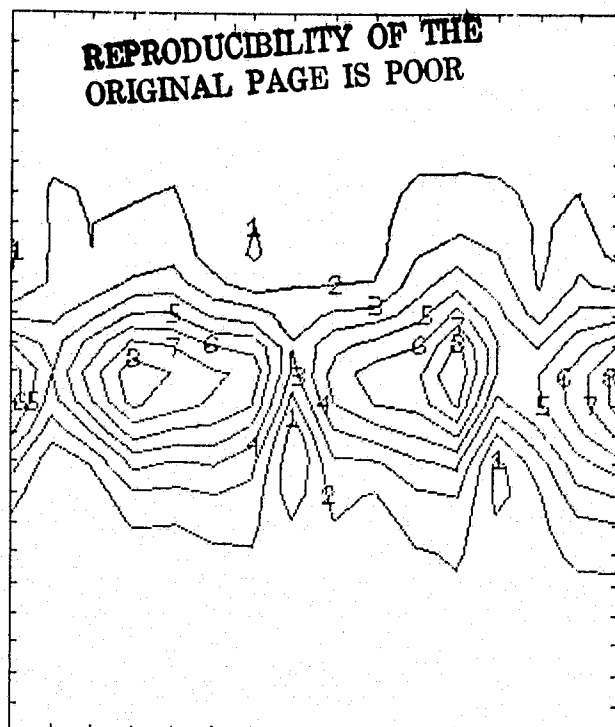


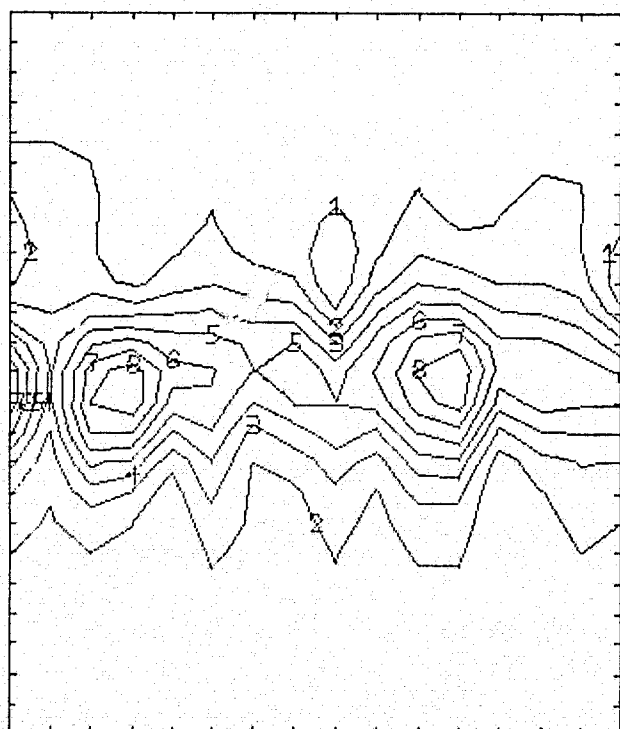
Fig. 6.9a. Contour plots of the spanwise vorticity $\bar{\omega}_2$ in an x - z plane, at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\omega_{2,\max} = 0.228$, case a).



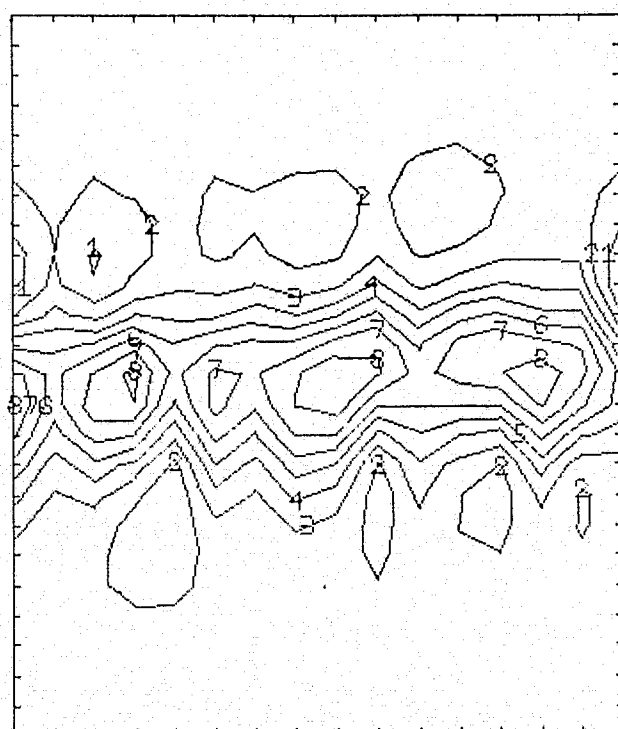
$$\frac{y}{\theta_{in}} = 0 ; \bar{\omega}_{2,max} = 0.314$$



$$\frac{y}{\theta_{in}} = 1.023 ; \bar{\omega}_{2,max} = 0.331$$

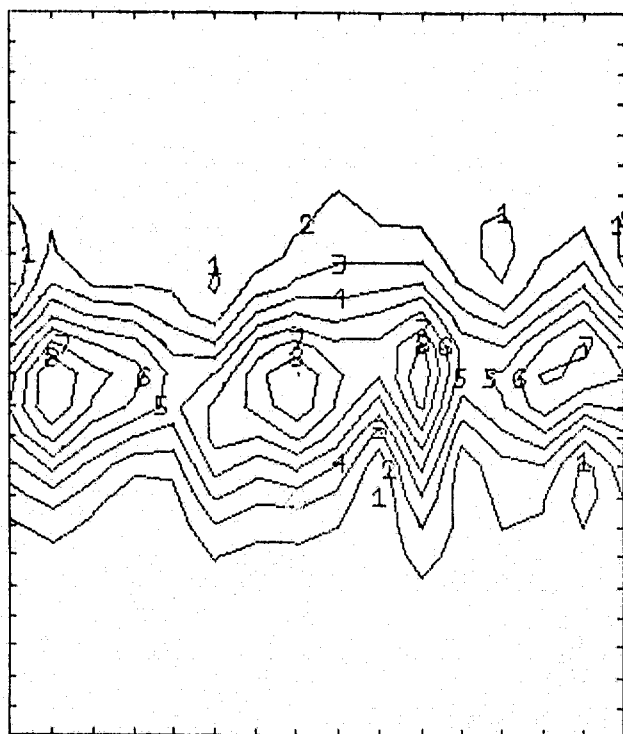


$$\frac{y}{\theta_{in}} = 2.026 ; \bar{\omega}_{2,max} = 0.376$$

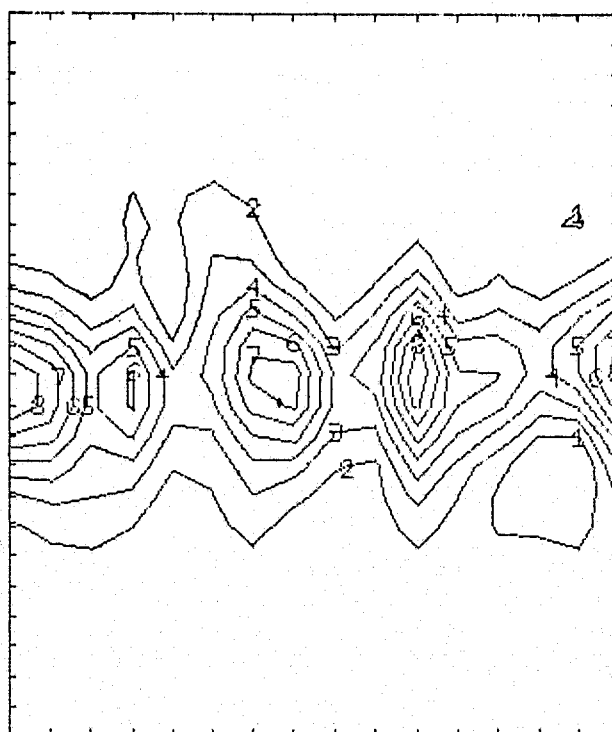


$$\frac{y}{\theta_{in}} = 3.069 ; \bar{\omega}_{2,max} = 0.321$$

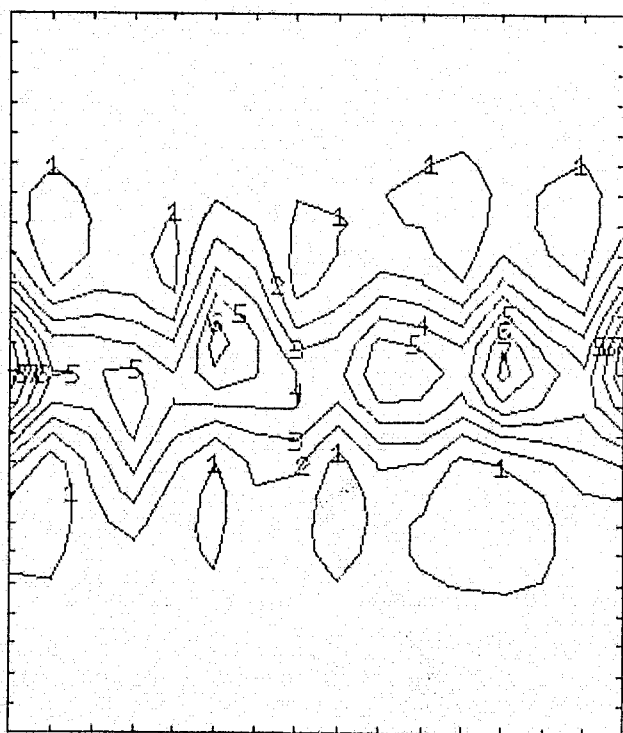
Figs. 6.9b. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x - z planes, at time $T = 0$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels (case b).



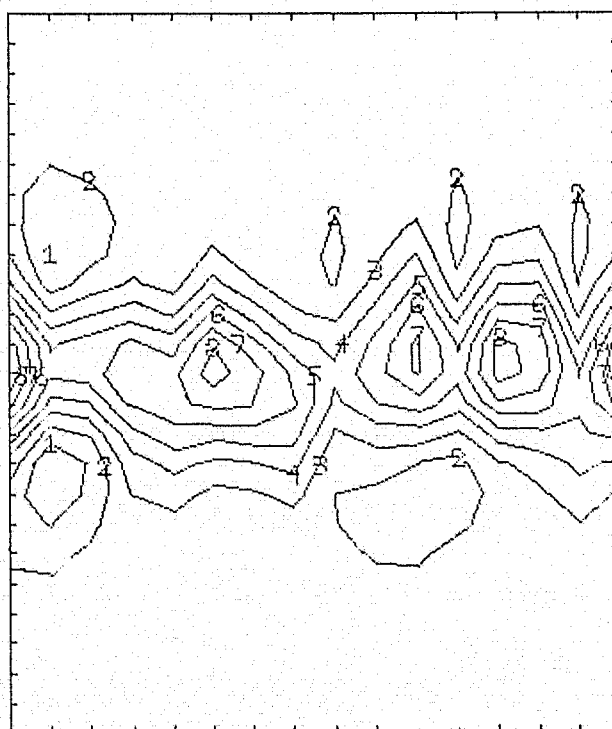
$$\frac{y}{\theta_{in}} = 4.092 ; \bar{\omega}_{2,max} = 0.316$$



$$\frac{y}{\theta_{in}} = 5.115 ; \bar{\omega}_{2,max} = 0.382$$



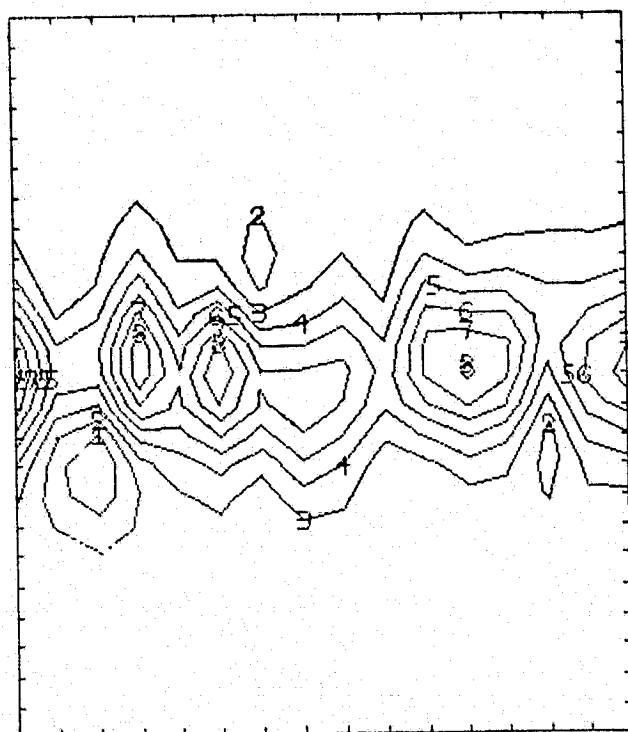
$$\frac{y}{\theta_{in}} = 6.138 ; \bar{\omega}_{2,max} = 0.432$$



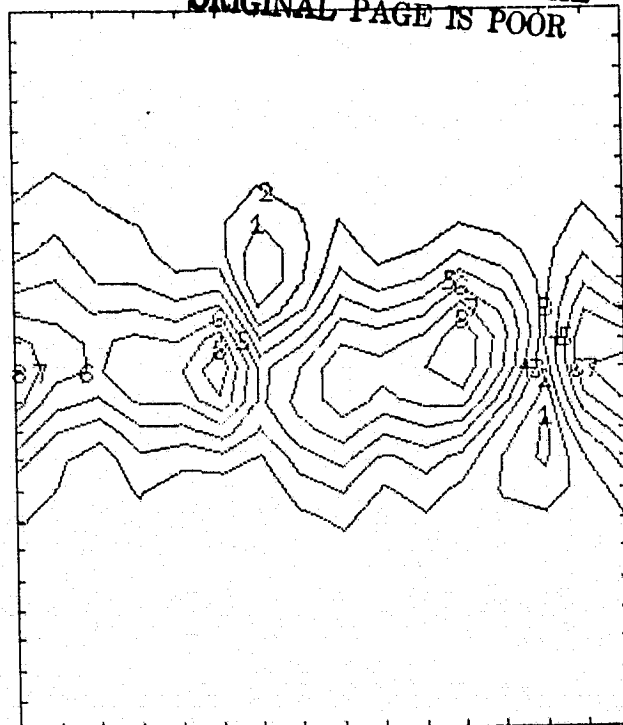
$$\frac{y}{\theta_{in}} = 7.161 ; \bar{\omega}_{2,max} = 0.363$$

Figs. 6.9b (continued)

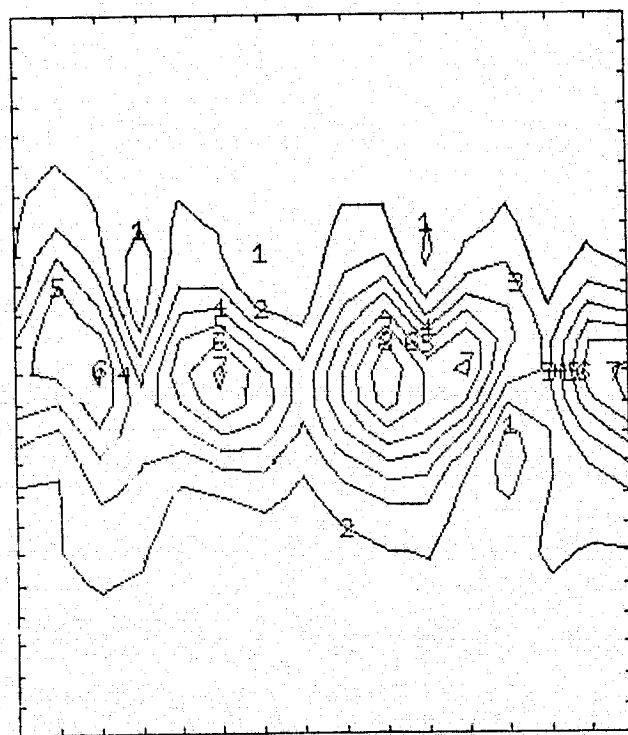
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR



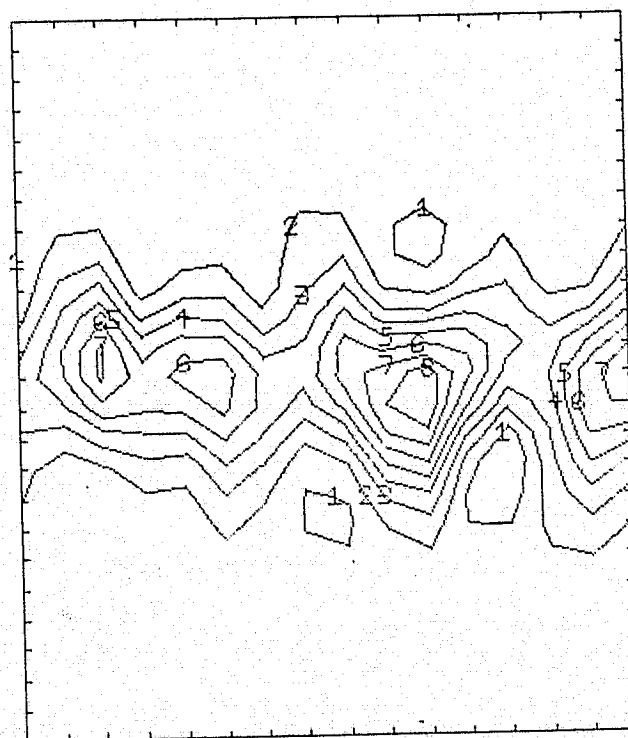
$$\frac{y}{\theta_{in}} = 8.184 ; \bar{\omega}_{2,max} = 0.386$$



$$\frac{y}{\theta_{in}} = 9.207 ; \bar{\omega}_{2,max} = 0.352$$

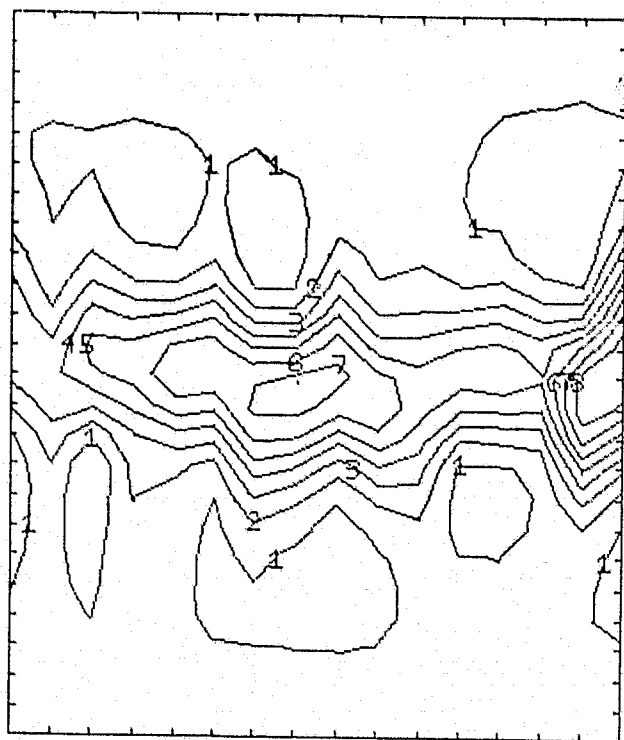


$$\frac{y}{\theta_{in}} = 10.230 ; \bar{\omega}_{2,max} = 0.426$$

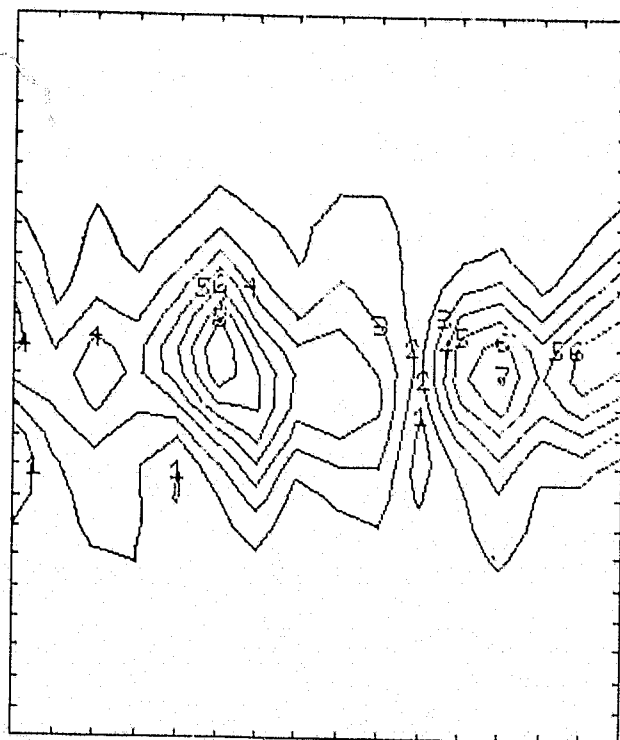


$$\frac{y}{\theta_{in}} = 11.253 ; \bar{\omega}_{2,max} = 0.382$$

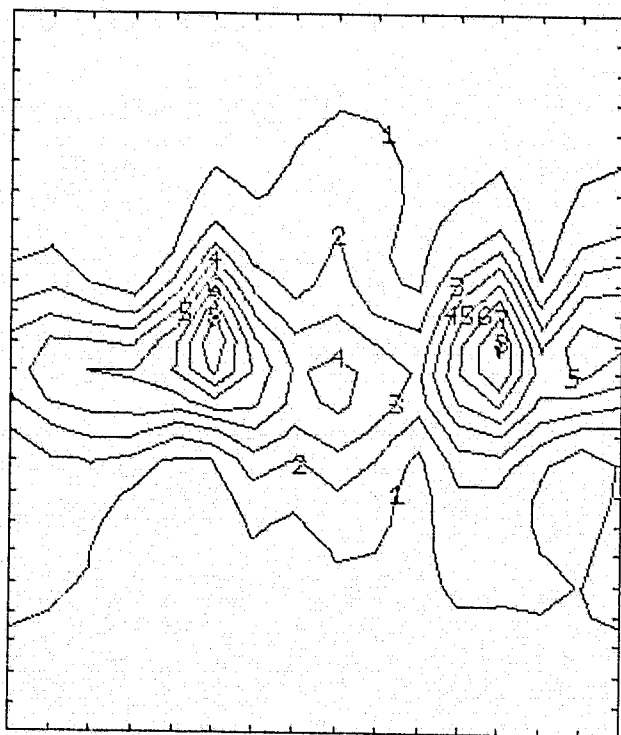
Figs. 6.9b (continued)



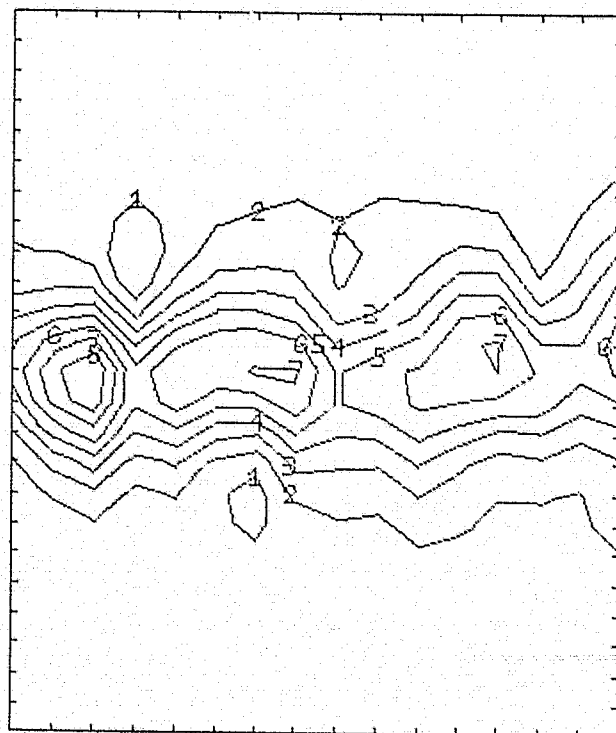
$$\frac{y}{\theta_{in}} = 12.276 \quad ; \quad \bar{\omega}_{2,max} = 0.341$$



$$\frac{y}{\theta_{in}} = 13.299 \quad ; \quad \bar{\omega}_{2,max} = 0.462$$

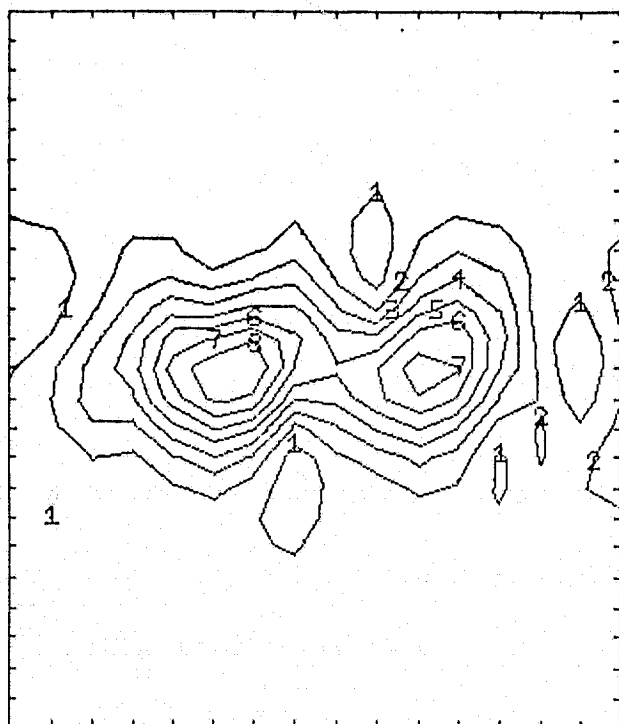


$$\frac{y}{\theta_{in}} = 14.322 \quad ; \quad \bar{\omega}_{2,max} = 0.427$$

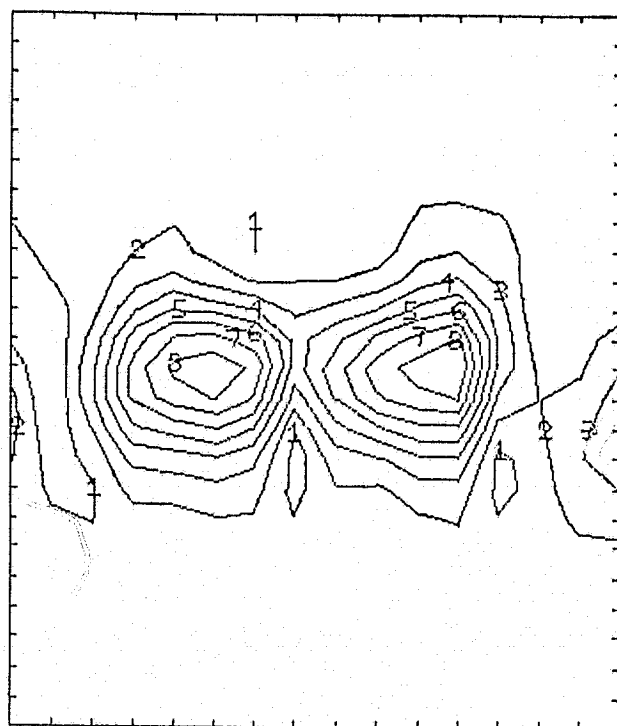


$$\frac{y}{\theta_{in}} = 15.345 \quad ; \quad \bar{\omega}_{2,max} = 0.327$$

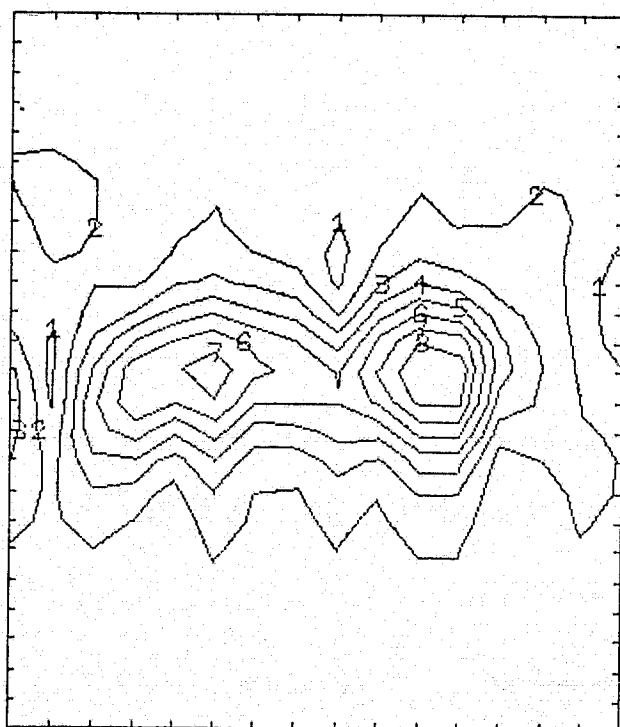
Figs. 6.9b (continued)



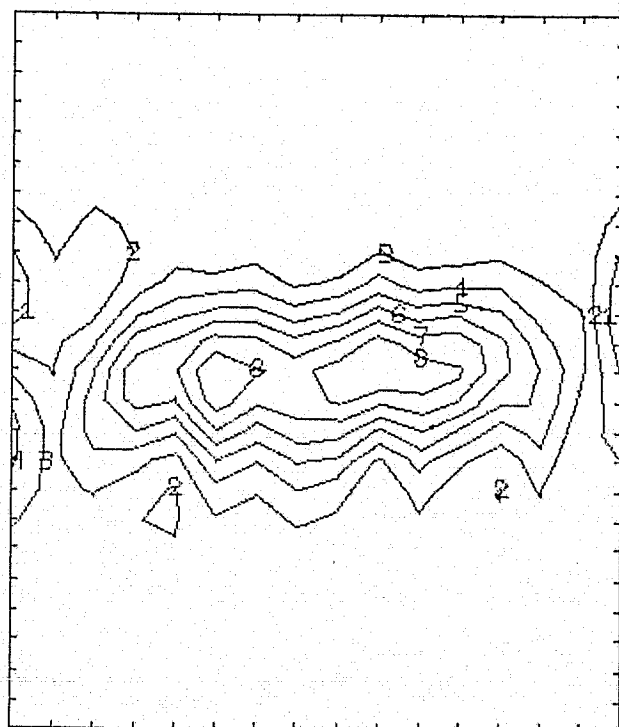
$$\frac{y}{\theta_{in}} = 0 ; \bar{\omega}_{2,max} = 0.468$$



$$\frac{y}{\theta_{in}} = 1.073 ; \bar{\omega}_{2,max} = 0.458$$

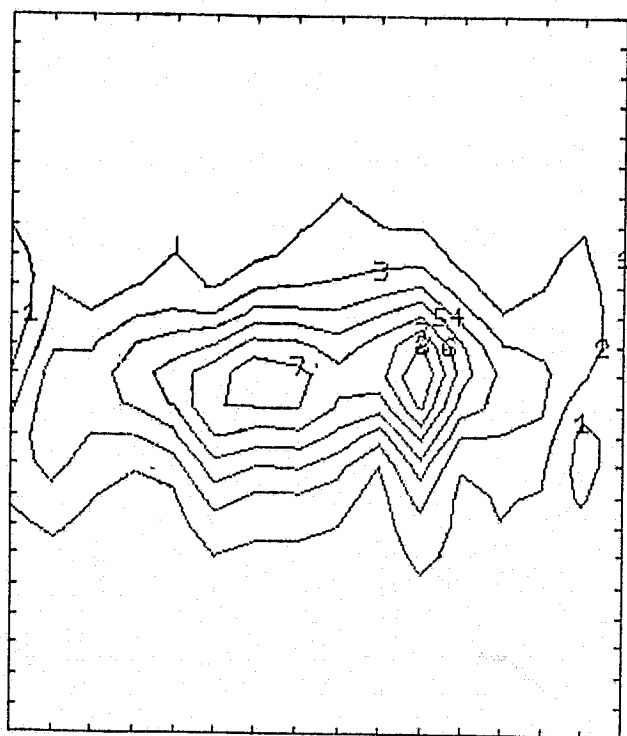


$$\frac{y}{\theta_{in}} = 2.026 ; \bar{\omega}_{2,max} = 0.523$$

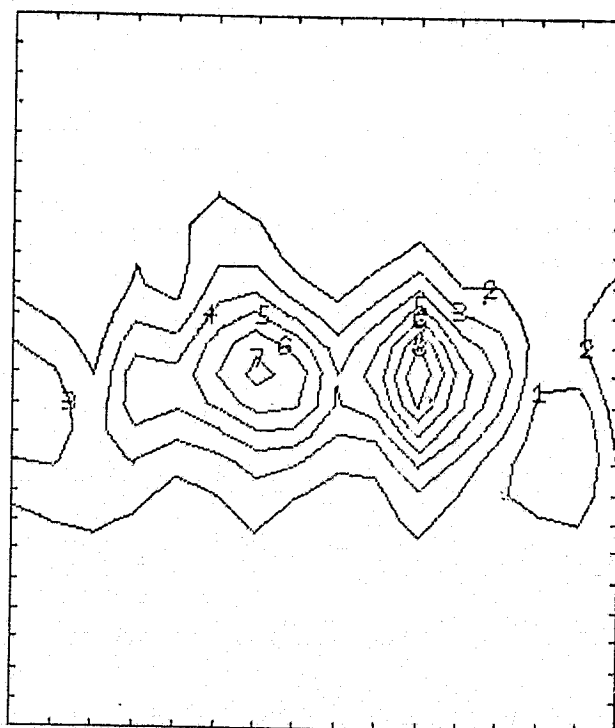


$$\frac{y}{\theta_{in}} = 3.069 ; \bar{\omega}_{2,max} = 0.436$$

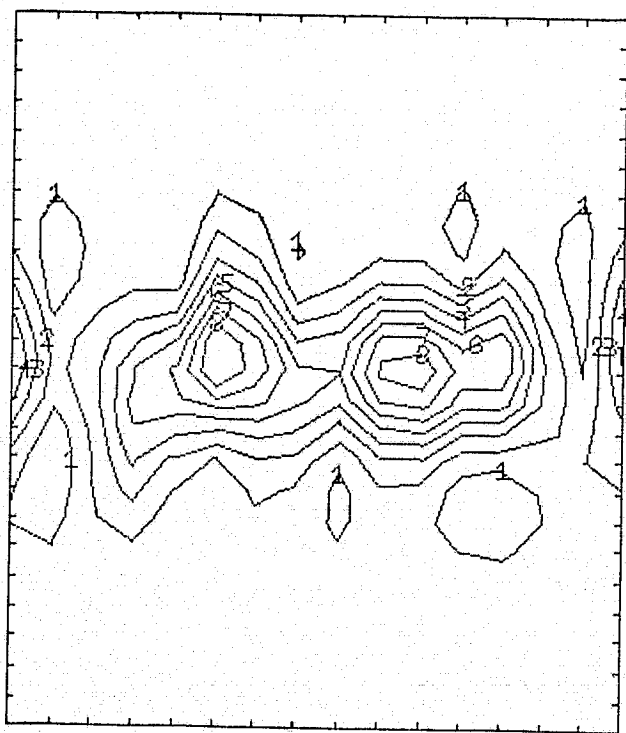
Figs. 6.9c. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x - z planes, at time $T = 0$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels (case c).



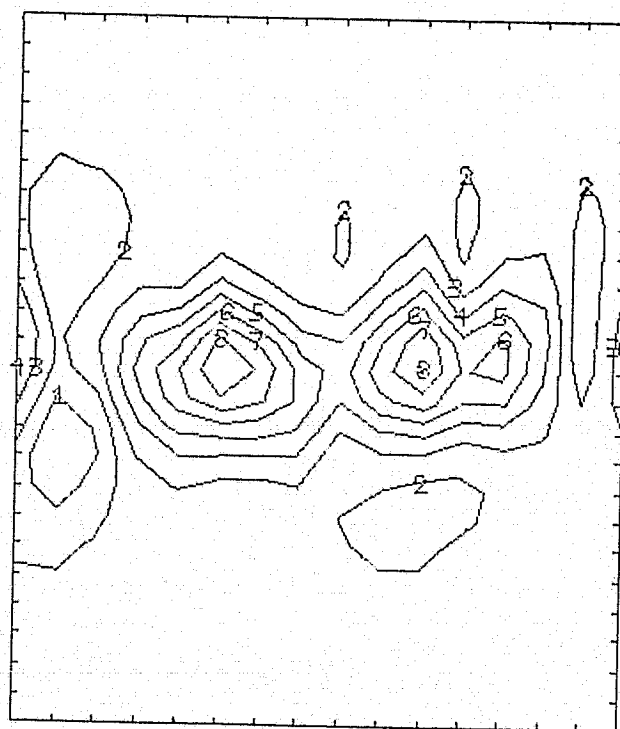
$$\frac{y}{\theta_{in}} = 4.092 \quad ; \quad \bar{\omega}_{2,max} = 0.507$$



$$\frac{y}{\theta_{in}} = 5.115 \quad ; \quad \bar{\omega}_{2,max} = 0.578$$

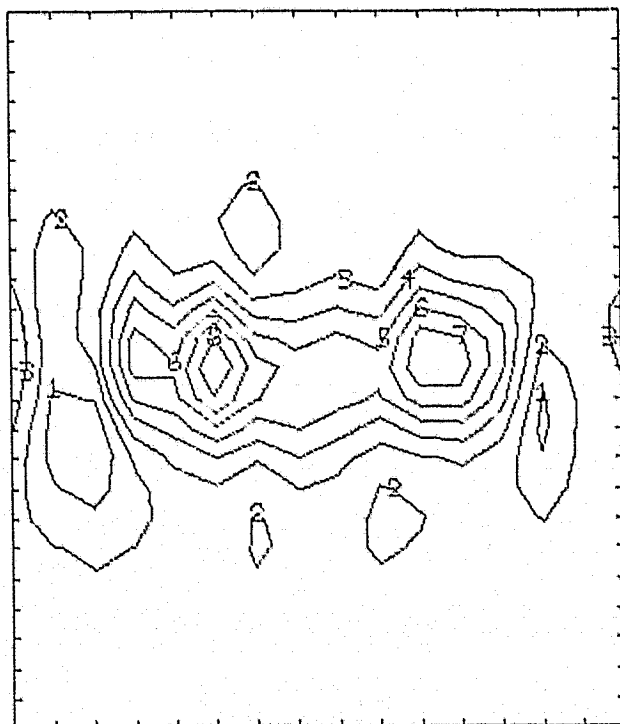


$$\frac{y}{\theta_{in}} = 6.138 \quad ; \quad \bar{\omega}_{2,max} = 0.457$$

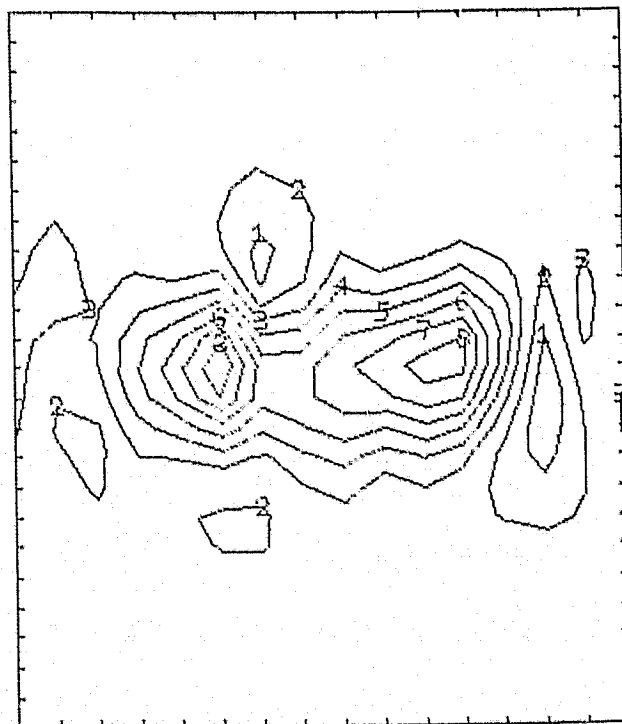


$$\frac{y}{\theta_{in}} = 7.161 \quad ; \quad \bar{\omega}_{2,max} = 0.529$$

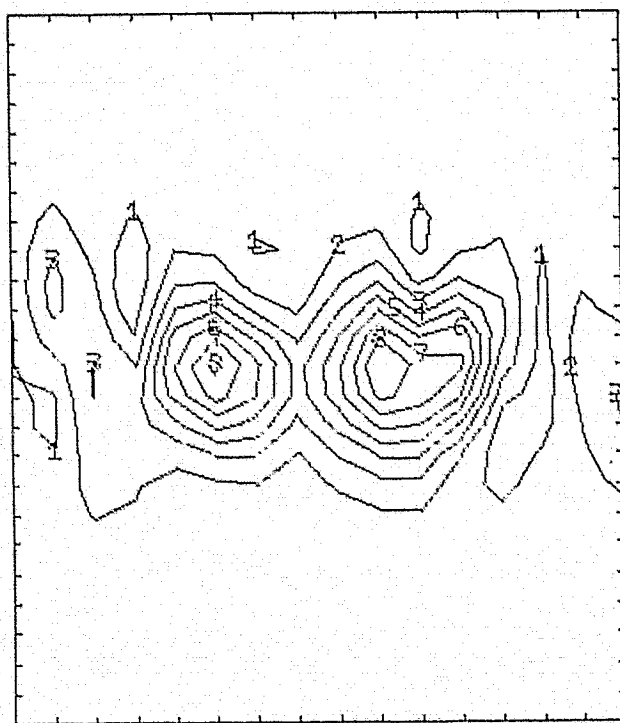
Figs. 6.9c (continued)



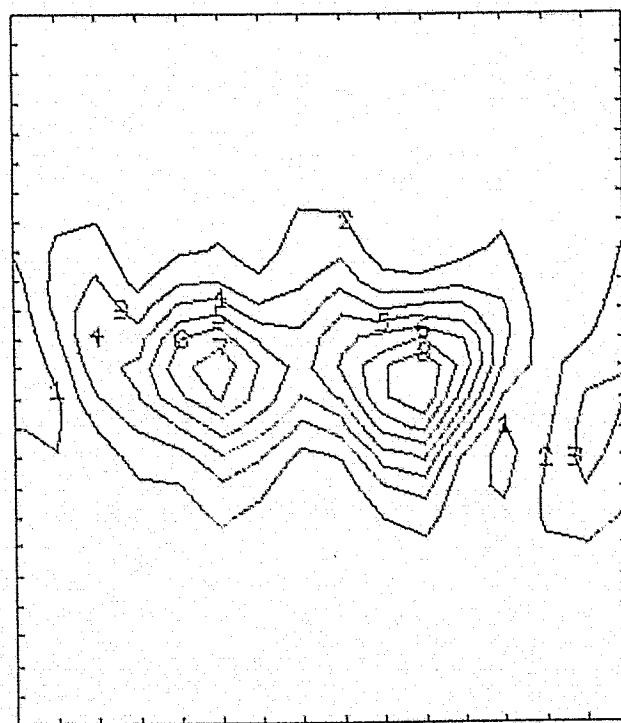
$$\frac{y}{\theta_{in}} = 8.184 \quad ; \quad \bar{\omega}_{2,max} = 0.582$$



$$\frac{y}{\theta_{in}} = 9.207 \quad ; \quad \bar{\omega}_{2,max} = 0.536$$

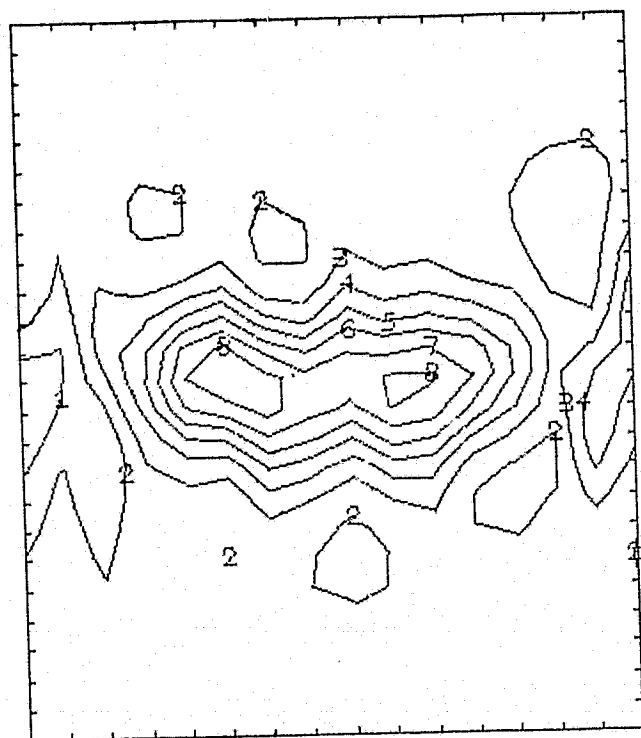


$$\frac{y}{\theta_{in}} = 10.230 \quad ; \quad \bar{\omega}_{2,max} = 0.580$$

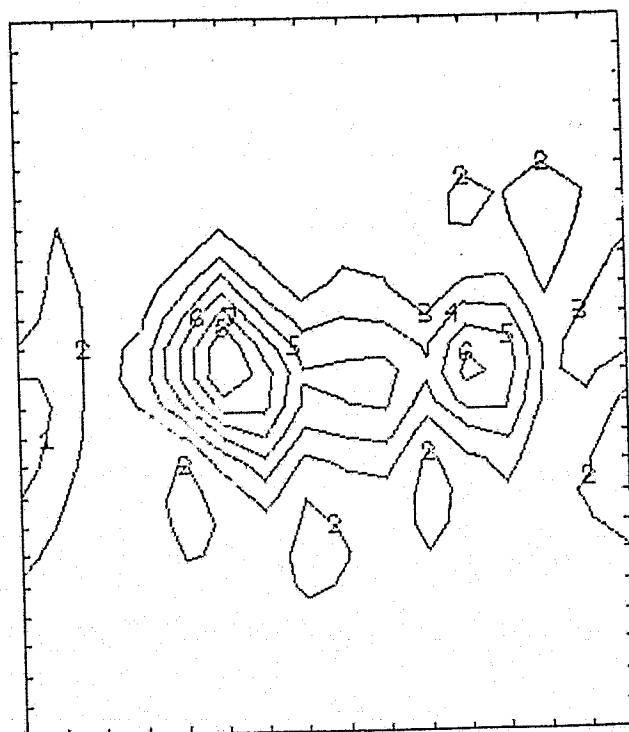


$$\frac{y}{\theta_{in}} = 11.253 \quad ; \quad \bar{\omega}_{2,max} = 0.549$$

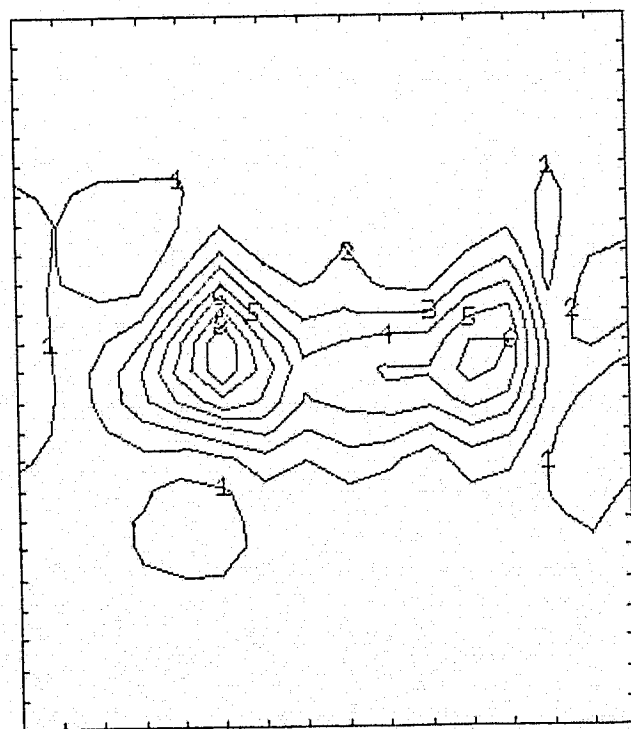
Figs. 6.9c (continued)



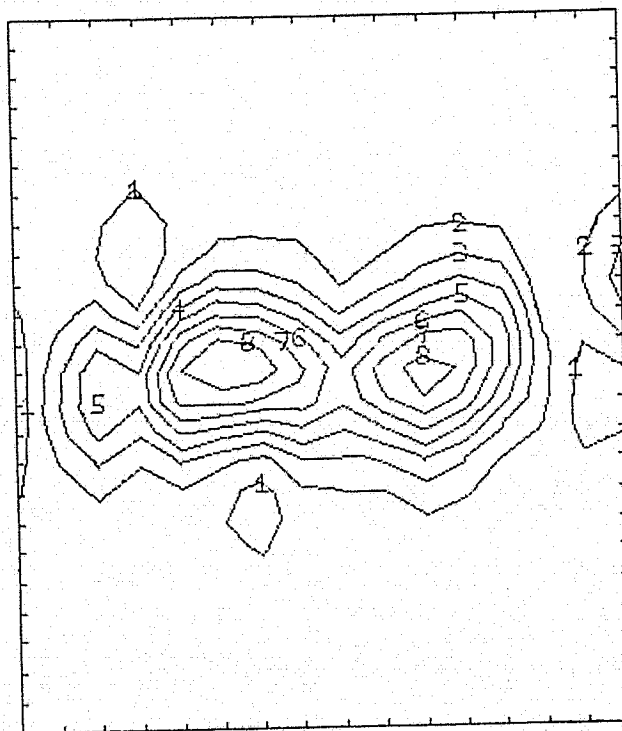
$$\frac{y}{\theta_{in}} = 12.276 \quad ; \quad \bar{\omega}_{2,max} = 0.441$$



$$\frac{y}{\theta_{in}} = 13.299 \quad ; \quad \bar{\omega}_{2,max} = 0.632$$



$$\frac{y}{\theta_{in}} = 14.322 \quad ; \quad \bar{\omega}_{2,max} = 0.593$$



$$\frac{y}{\theta_{in}} = 15.345 \quad ; \quad \bar{\omega}_{2,max} = 0.422$$

Figs. 6.9c (continued)

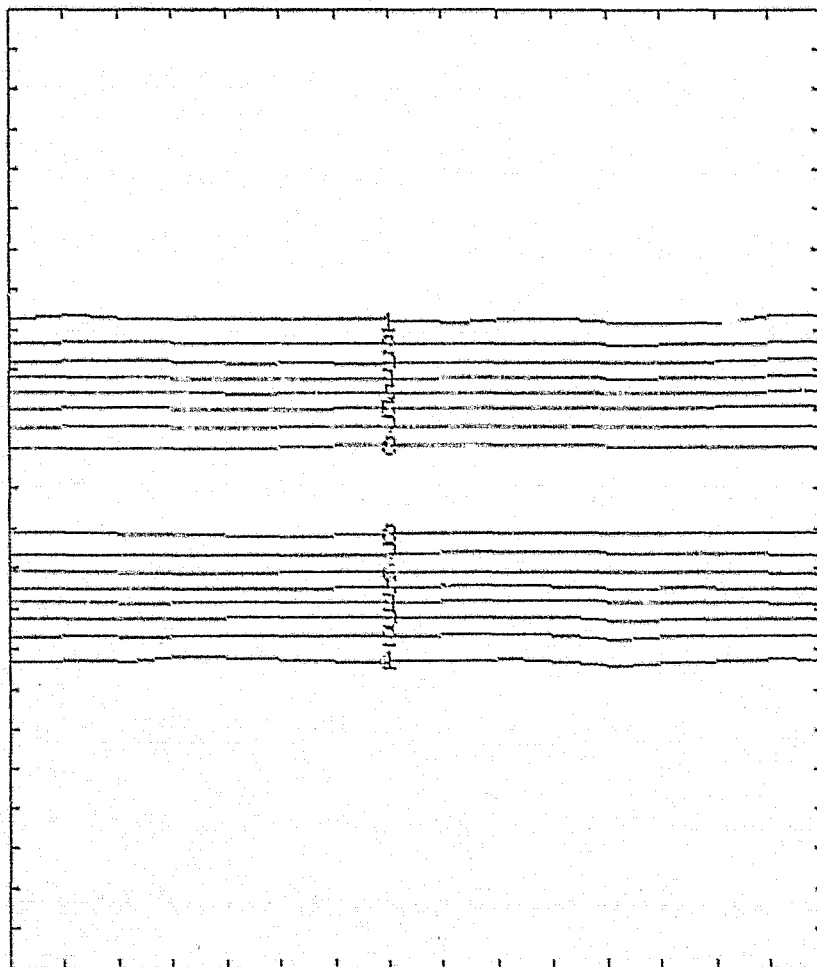
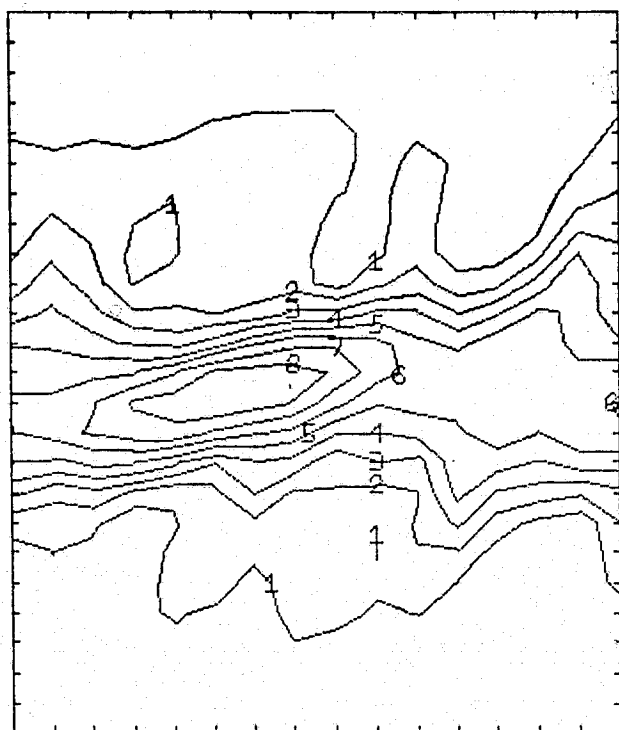
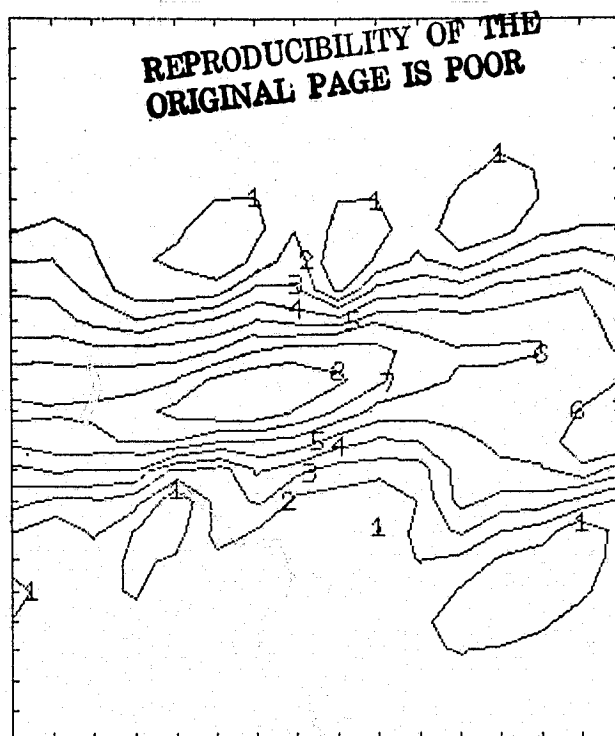


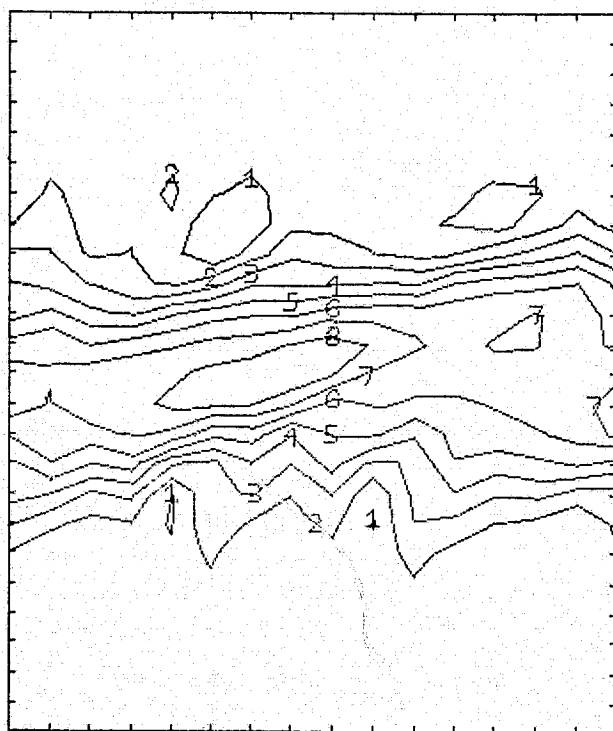
Fig. 6.10a. Contour plots of the spanwise vorticity $(\bar{\omega}_2)$ in an x-z plane, at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\omega_{2,\max} = 0.185$, case a).



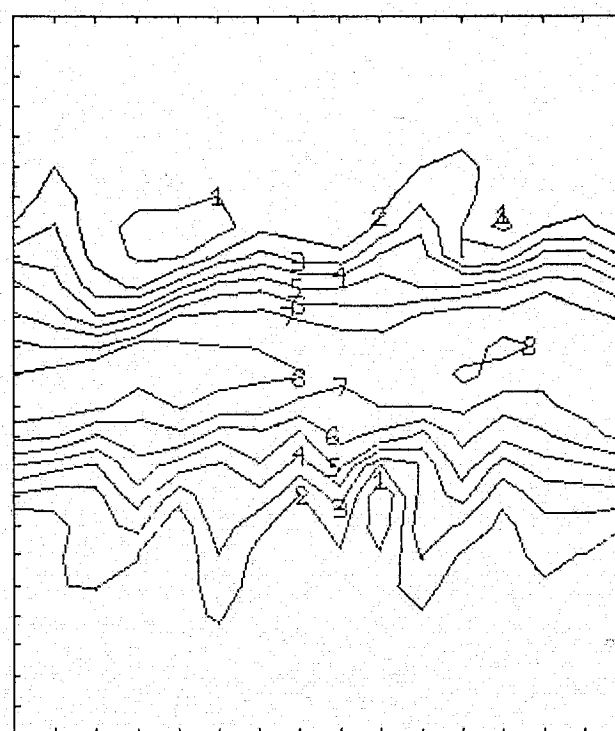
$$\frac{y}{\theta_{in}} = 0 ; \bar{\omega}_{2,max} = 0.255$$



$$\frac{y}{\theta_{in}} = 1.023 ; \bar{\omega}_{2,max} = 0.231$$

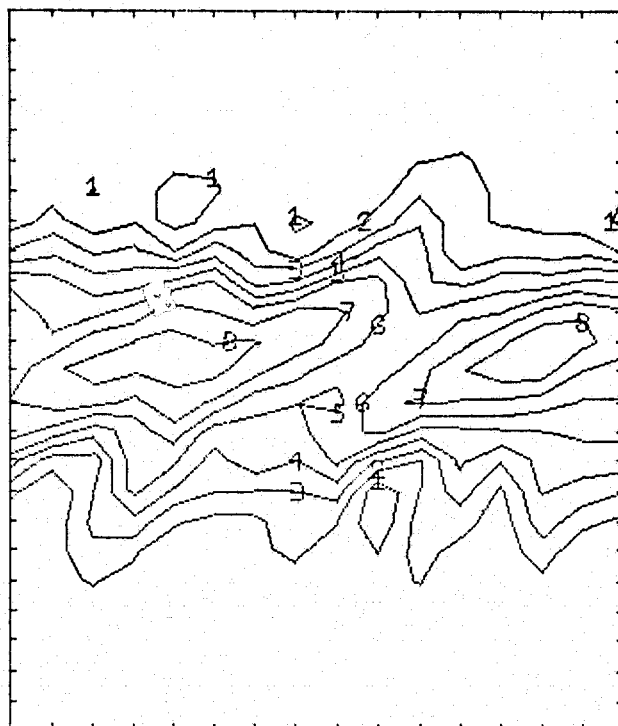


$$\frac{y}{\theta_{in}} = 2.026 ; \bar{\omega}_{2,max} = 0.224$$

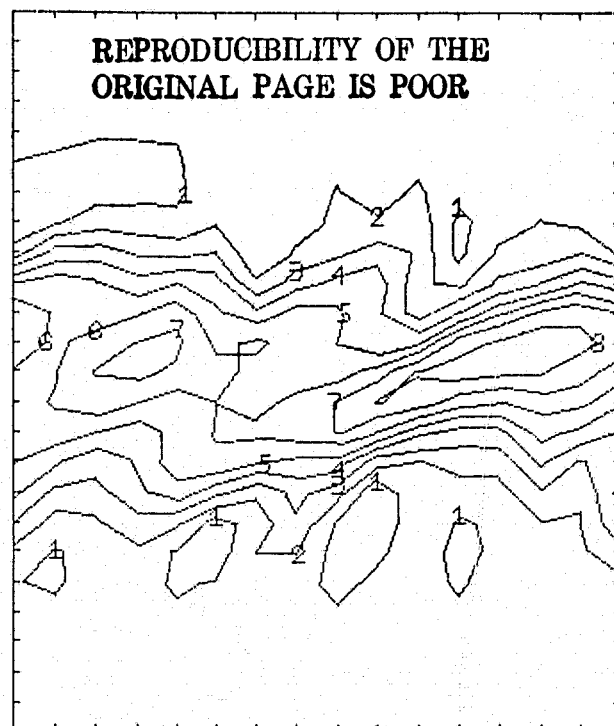


$$\frac{y}{\theta_{in}} = 3.069 ; \bar{\omega}_{2,max} = 0.203$$

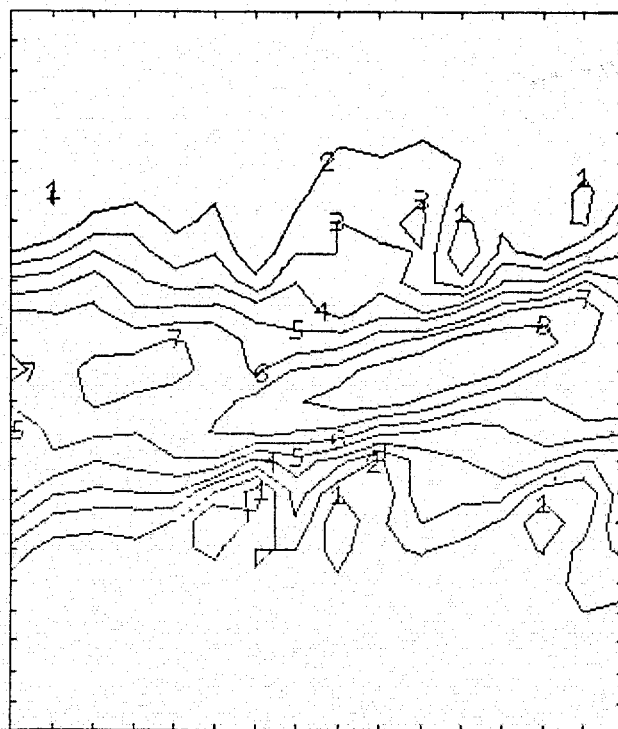
Fig. 6.10b. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x-z planes, at time $T = 16.78$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case b).



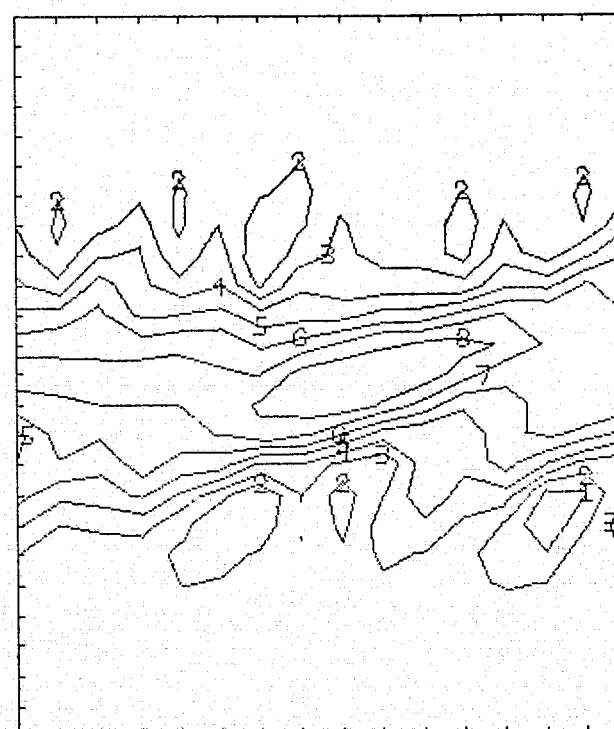
$$\frac{y}{\theta_{in}} = 4.092 \quad ; \quad \bar{\omega}_{2,max} = 0.209$$



$$\frac{y}{\theta_{in}} = 5.115 \quad ; \quad \bar{\omega}_{2,max} = 0.227$$

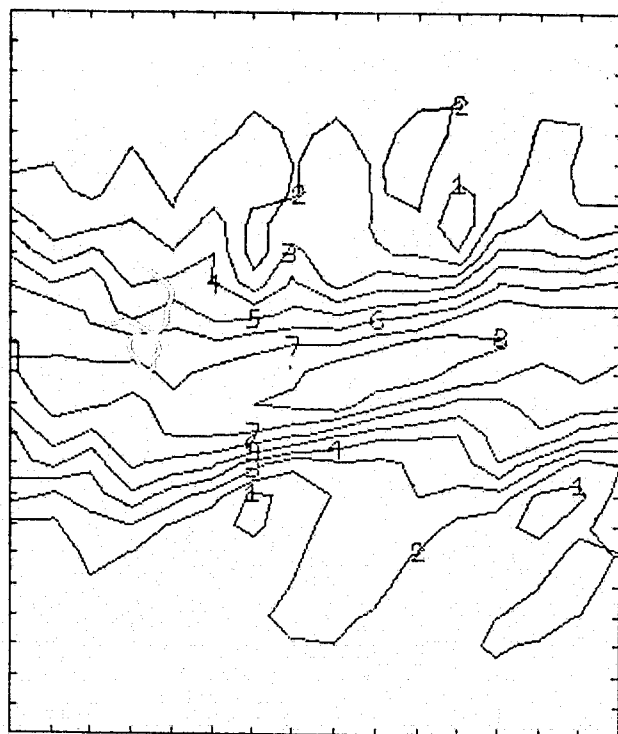


$$\frac{y}{\theta_{in}} = 6.138 \quad ; \quad \bar{\omega}_{2,max} = 0.228$$

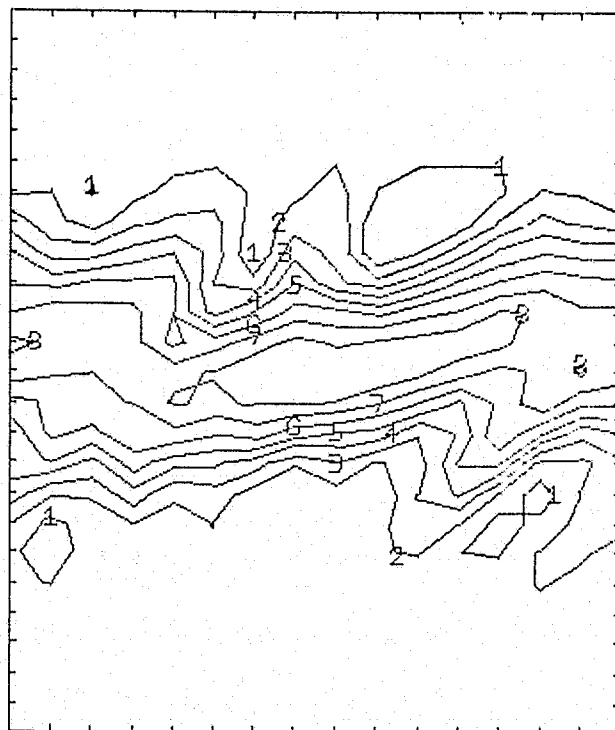


$$\frac{y}{\theta_{in}} = 7.161 \quad ; \quad \bar{\omega}_{2,max} = 0.244$$

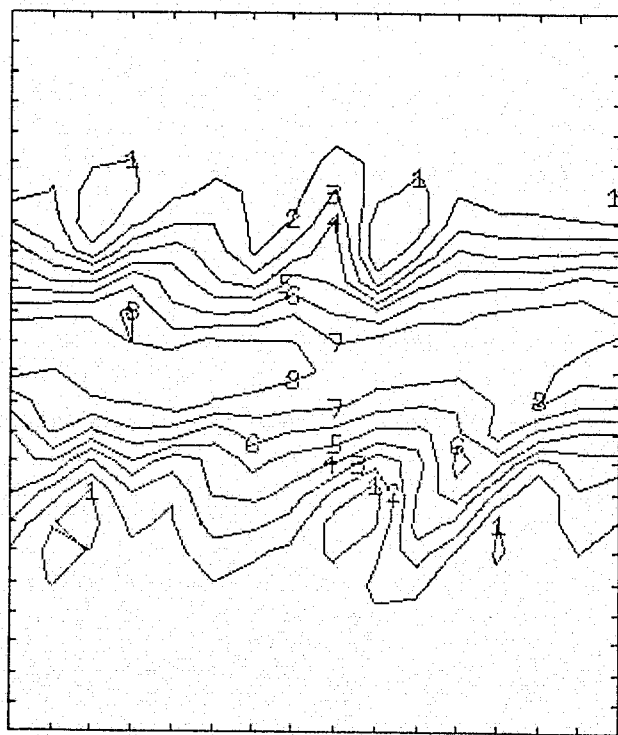
Figs. 6.10b (continued)



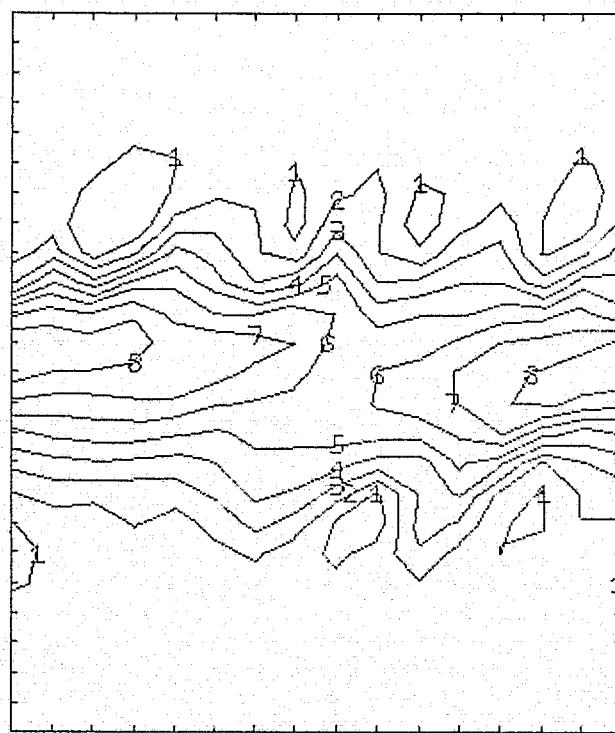
$$\frac{y}{\theta_{in}} = 8.184 ; \bar{\omega}_{2,max} = 0.231$$



$$\frac{y}{\theta_{in}} = 9.207 ; \bar{\omega}_{2,max} = 0.207$$

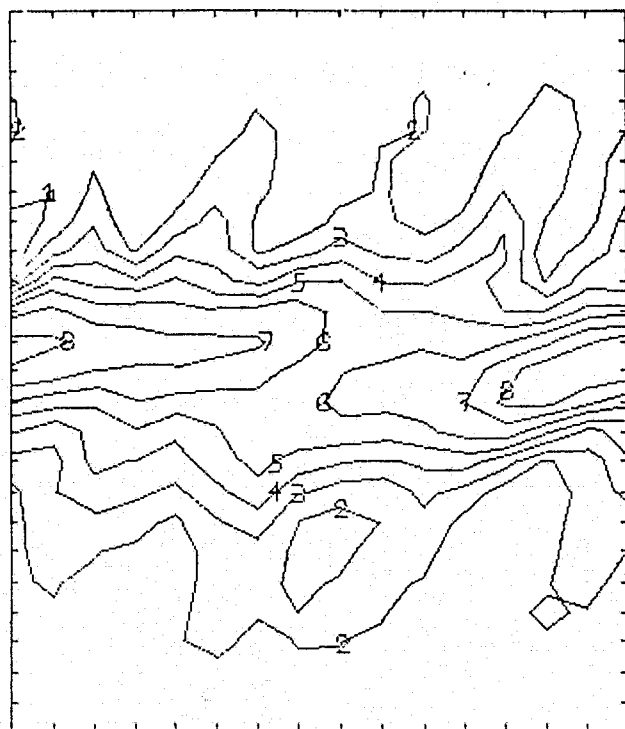


$$\frac{y}{\theta_{in}} = 10.230 ; \bar{\omega}_{2,max} = 0.191$$

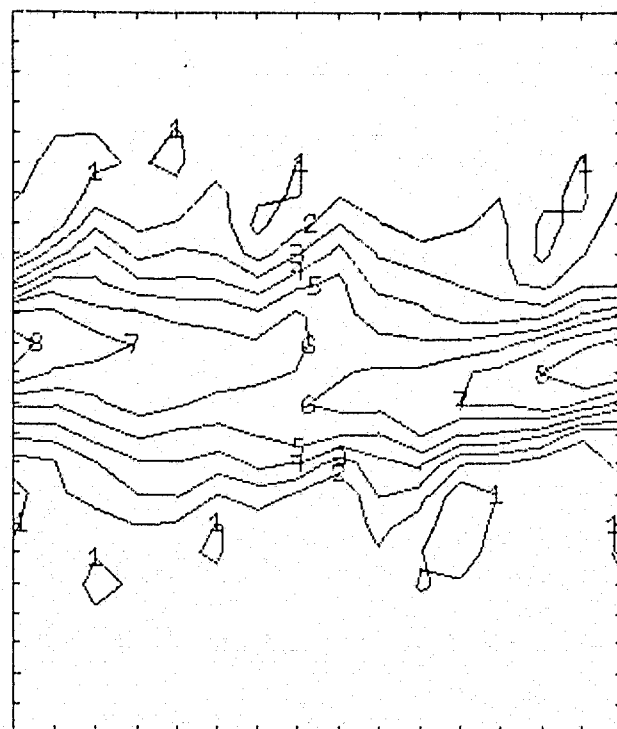


$$\frac{y}{\theta_{in}} = 11.253 ; \bar{\omega}_{2,max} = 0.221$$

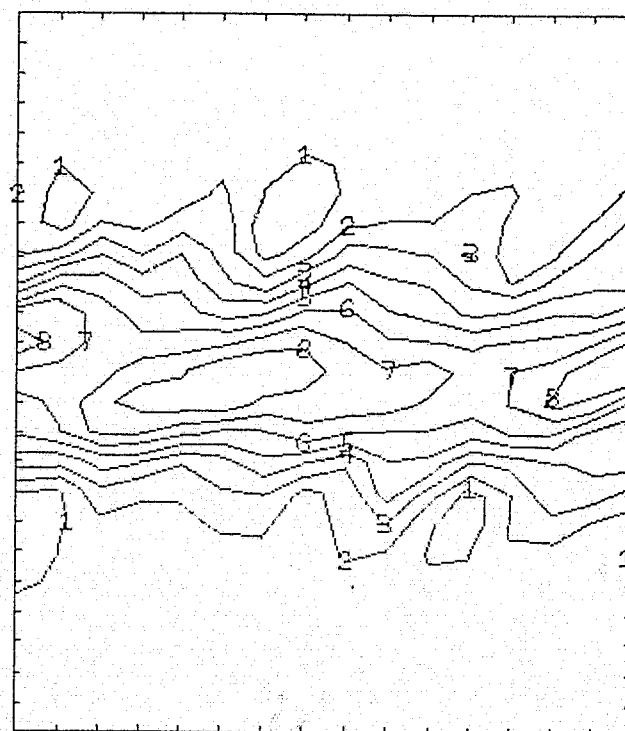
Figs. 6.10b (continued)



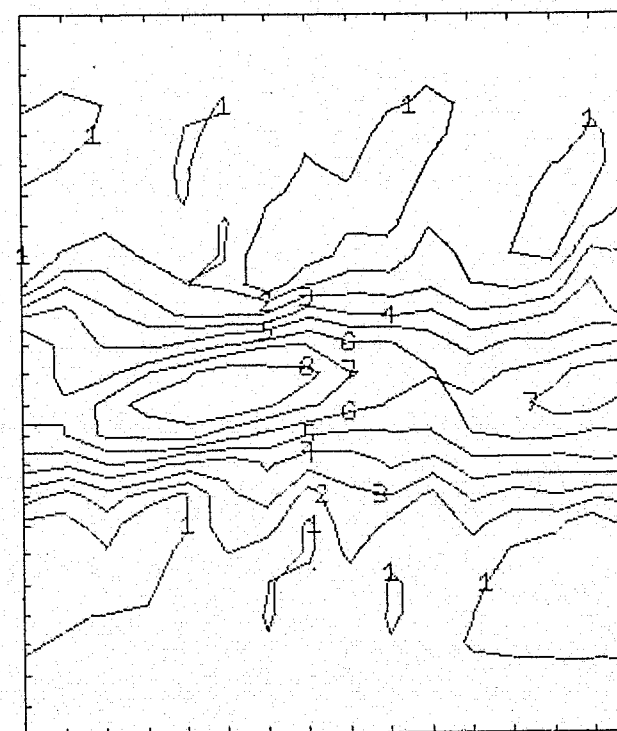
$$\frac{y}{\theta_{in}} = 12.276 \quad ; \quad \bar{\omega}_{2,max} = 0.258$$



$$\frac{y}{\theta_{in}} = 13.299 \quad ; \quad \bar{\omega}_{2,max} = 0.253$$

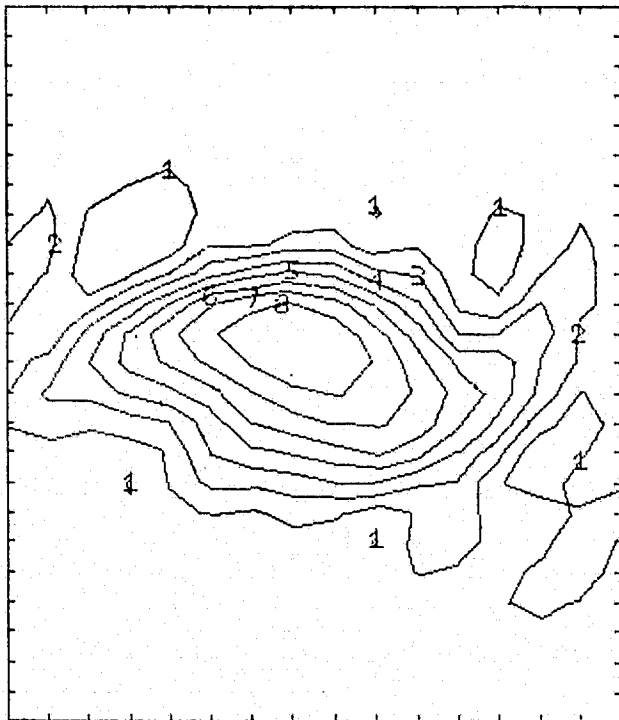


$$\frac{y}{\theta_{in}} = 12.322 \quad ; \quad \bar{\omega}_{2,max} = 0.216$$

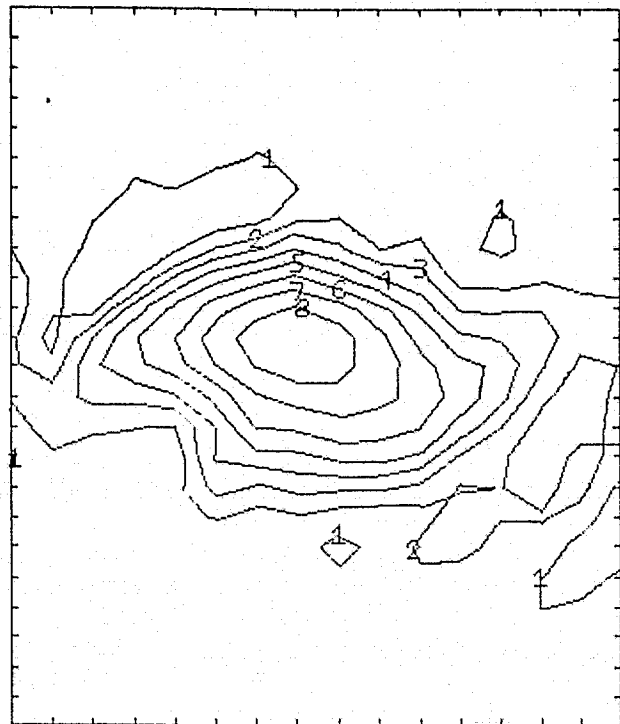


$$\frac{y}{\theta_{in}} = 15.345 \quad ; \quad \bar{\omega}_{2,max} = 0.244$$

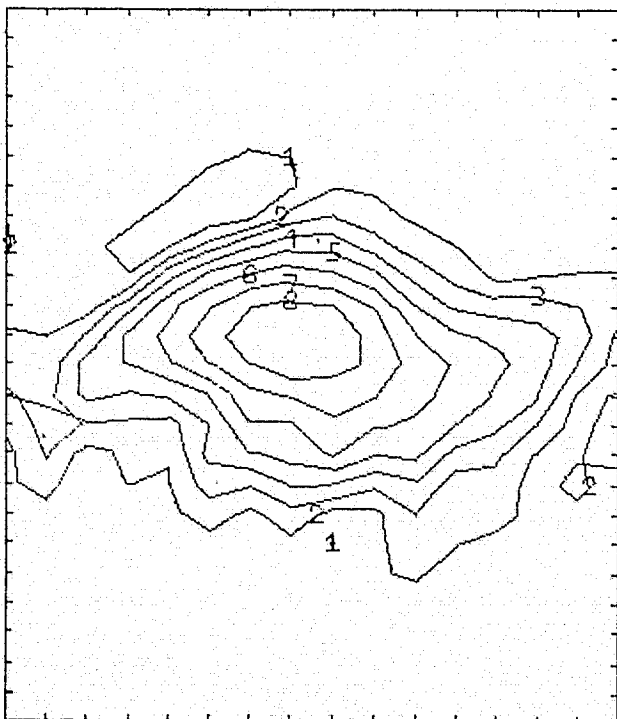
Figs. 6.10b (continued)



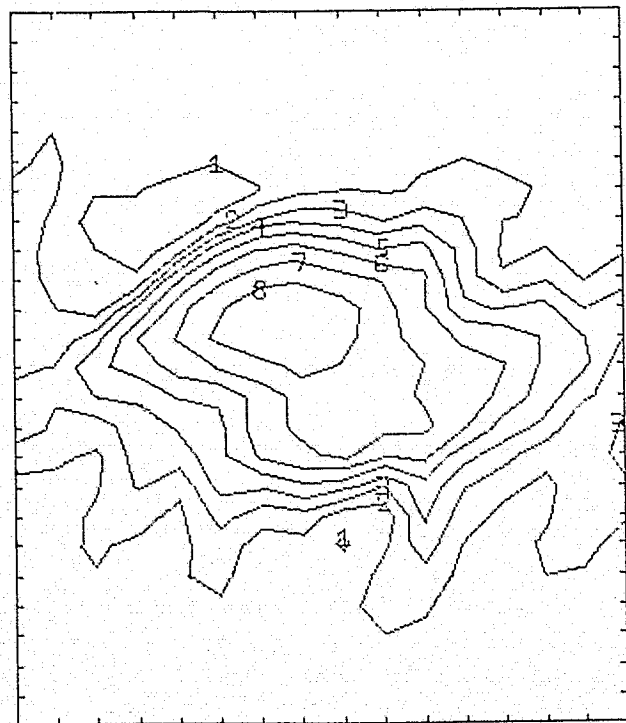
$$\frac{y}{\theta_{in}} = 0 ; \bar{\omega}_{2,max} = 0.337$$



$$\frac{y}{\theta_{in}} = 1.023 ; \bar{\omega}_{2,max} = 0.347$$

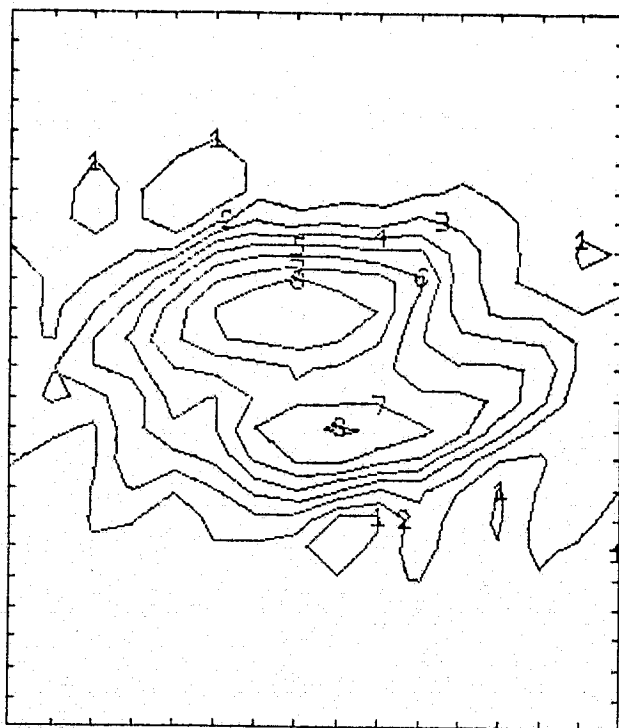


$$\frac{y}{\theta_{in}} = 2.026 ; \bar{\omega}_{2,max} = 0.335$$

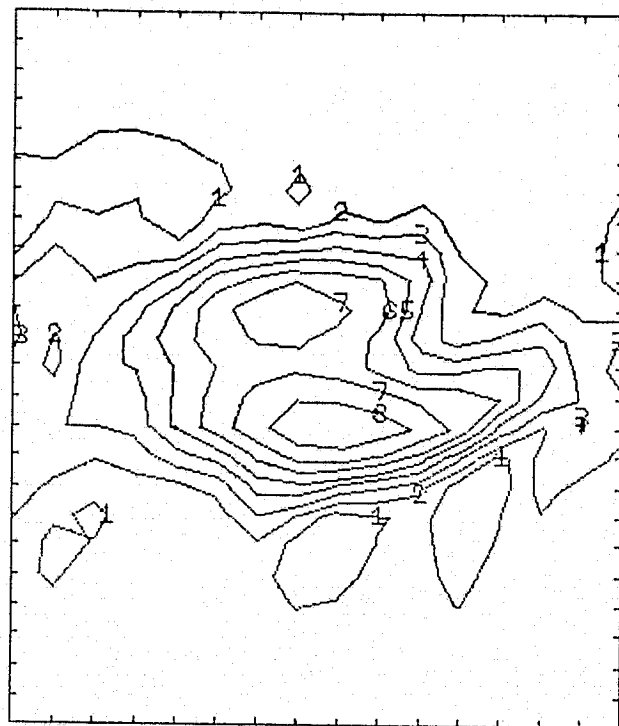


$$\frac{y}{\theta_{in}} = 3.069 ; \bar{\omega}_{2,max} = 0.286$$

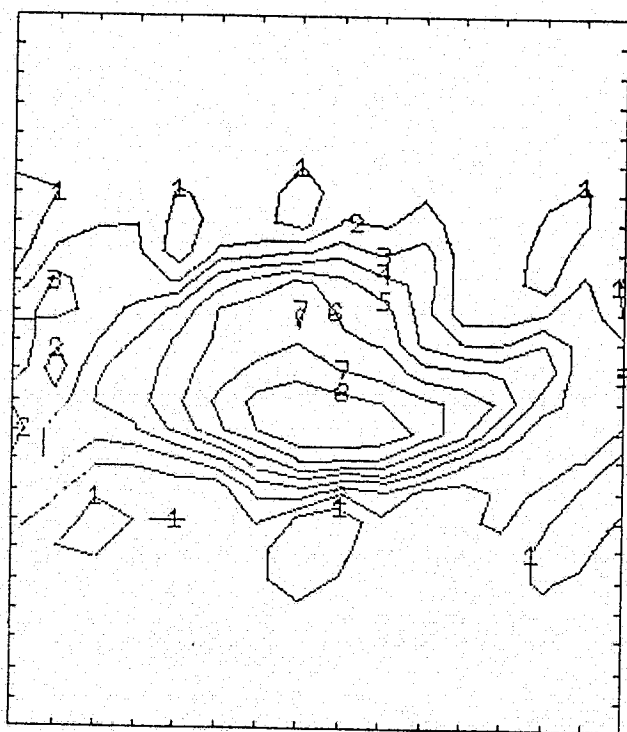
Figs. 6.10c. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) for different x - z planes, at time $T = 16.78$. In each plane, constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case c).



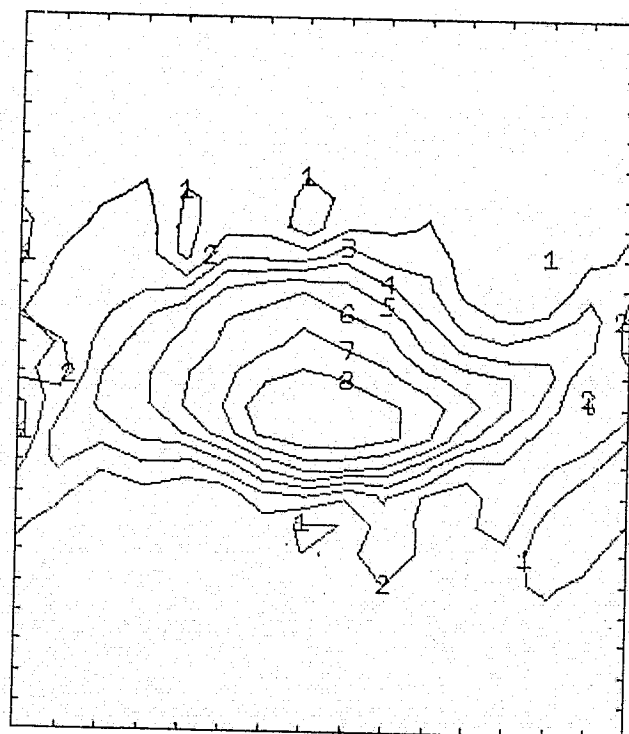
$$\frac{y}{\theta_{in}} = 4.092 ; \bar{\omega}_{2,max} = 0.280$$



$$\frac{y}{\theta_{in}} = 5.115 ; \bar{\omega}_{2,max} = 0.305$$

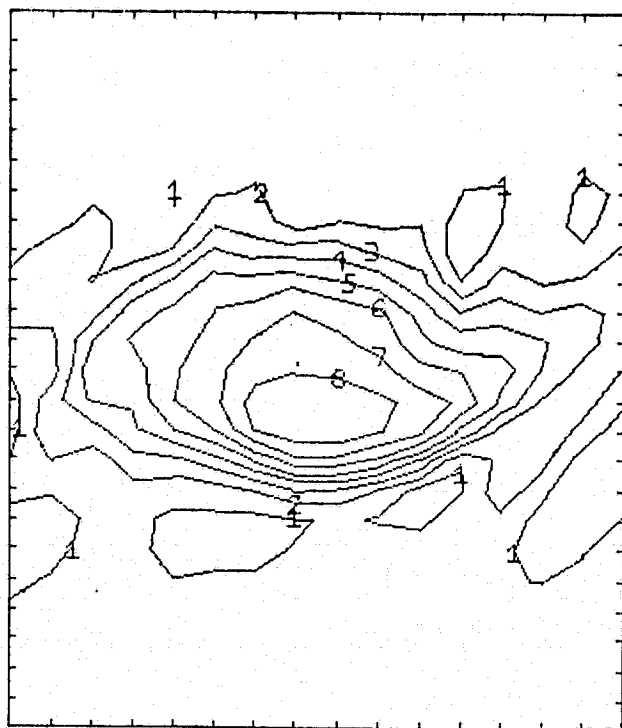


$$\frac{y}{\theta_{in}} = 6.138 ; \bar{\omega}_{2,max} = 0.335$$

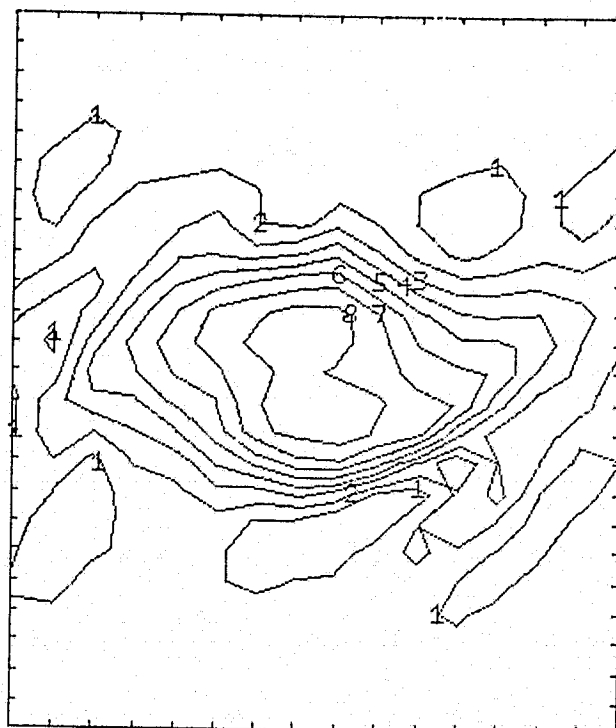


$$\frac{y}{\theta_{in}} = 7.161 ; \bar{\omega}_{2,max} = 0.344$$

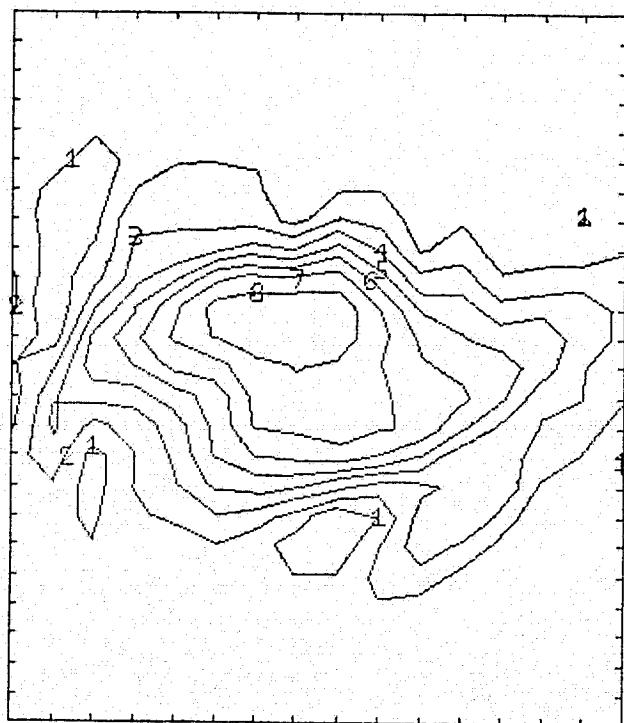
Figs. 6.10c (continued)



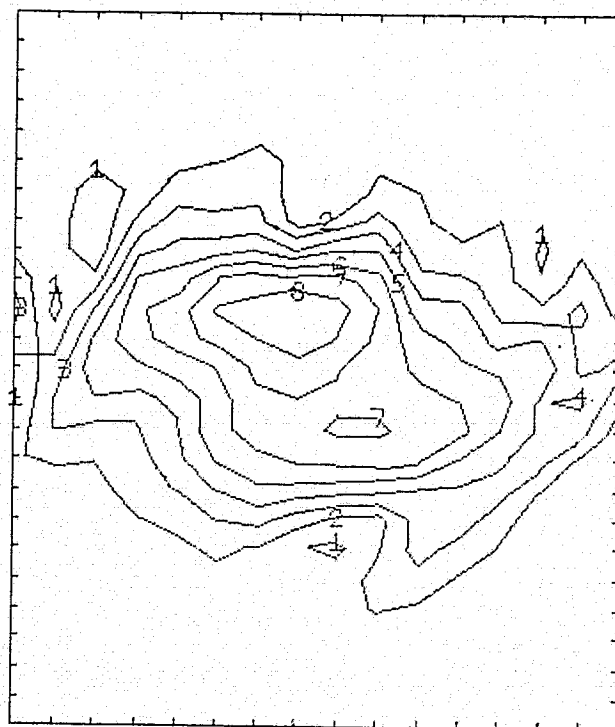
$$\frac{y}{\theta_{in}} = 8.184 \quad ; \quad \bar{\omega}_{2,max} = 0.322$$



$$\frac{y}{\theta_{in}} = 9.207 \quad ; \quad \bar{\omega}_{2,max} = 0.291$$

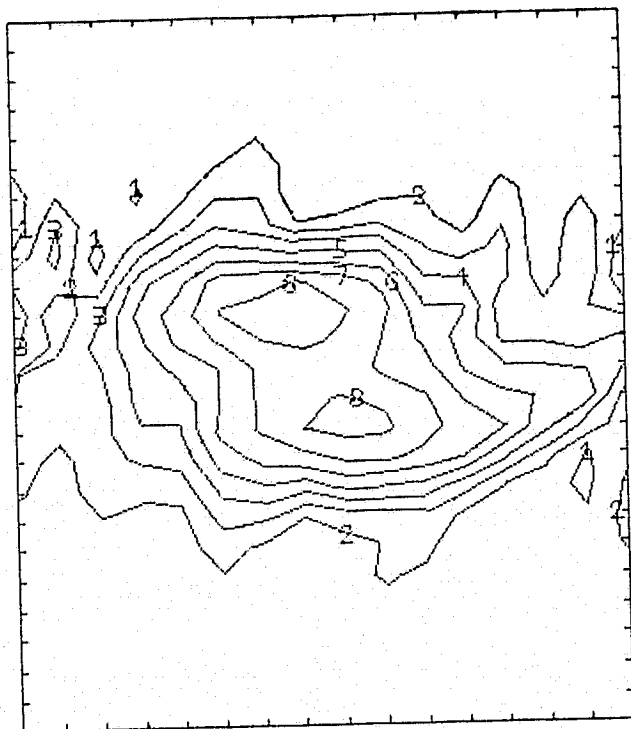


$$\frac{y}{\theta_{in}} = 10.230 \quad ; \quad \bar{\omega}_{2,max} = 0.286$$

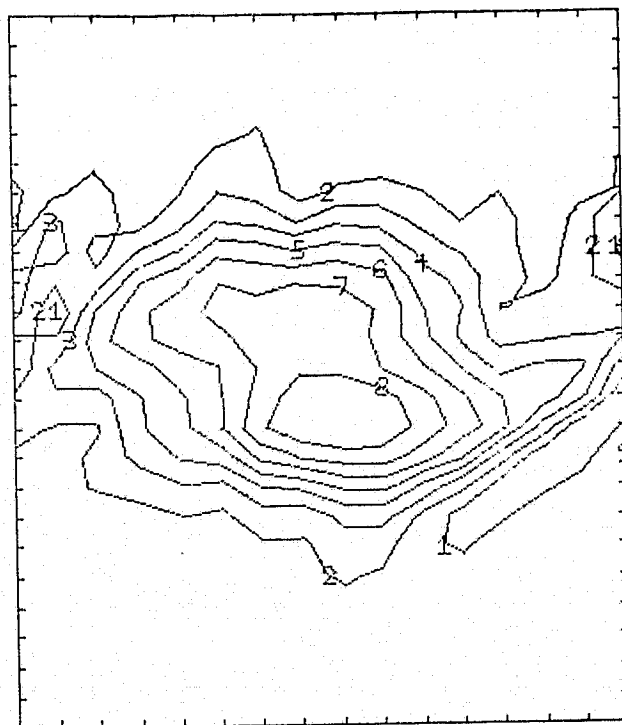


$$\frac{y}{\theta_{in}} = 11.253 \quad ; \quad \bar{\omega}_{2,max} = 0.292$$

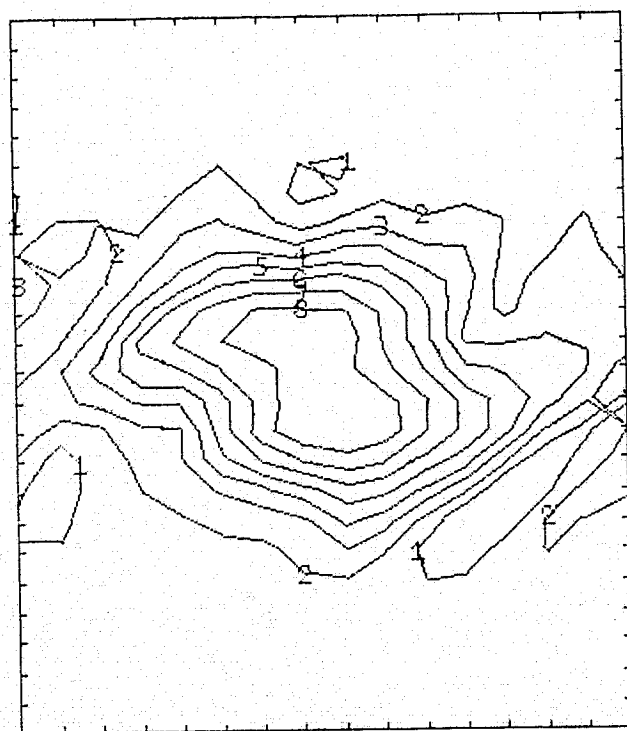
Figs. 6.10c (continued)



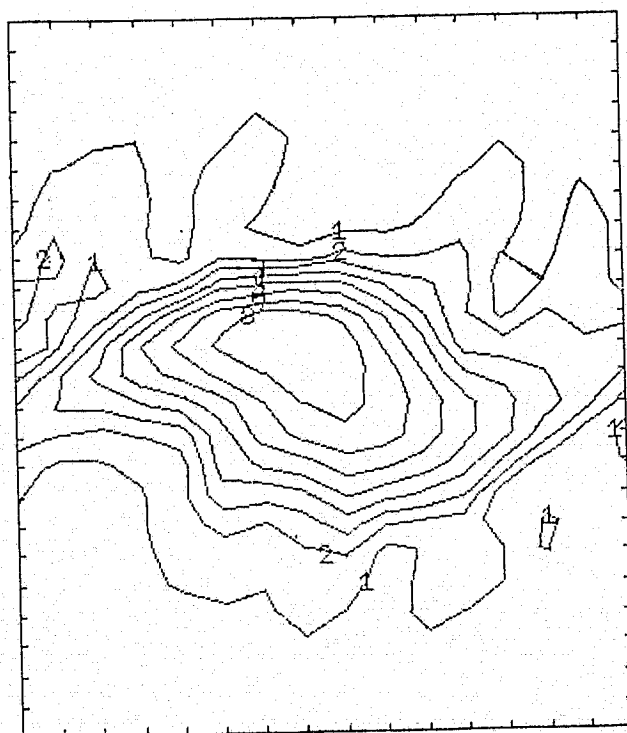
$$\frac{y}{\theta_{in}} = 12.276 \quad ; \quad \bar{\omega}_{2,max} = 0.279$$



$$\frac{y}{\theta_{in}} = 13.299 \quad ; \quad \bar{\omega}_{2,max} = 0.286$$



$$\frac{y}{\theta_{in}} = 14.322 \quad ; \quad \bar{\omega}_{2,max} = 0.293$$



$$\frac{y}{\theta_{in}} = 15.345 \quad ; \quad \bar{\omega}_{2,max} = 0.319$$

Figs. 6.10c (continued)

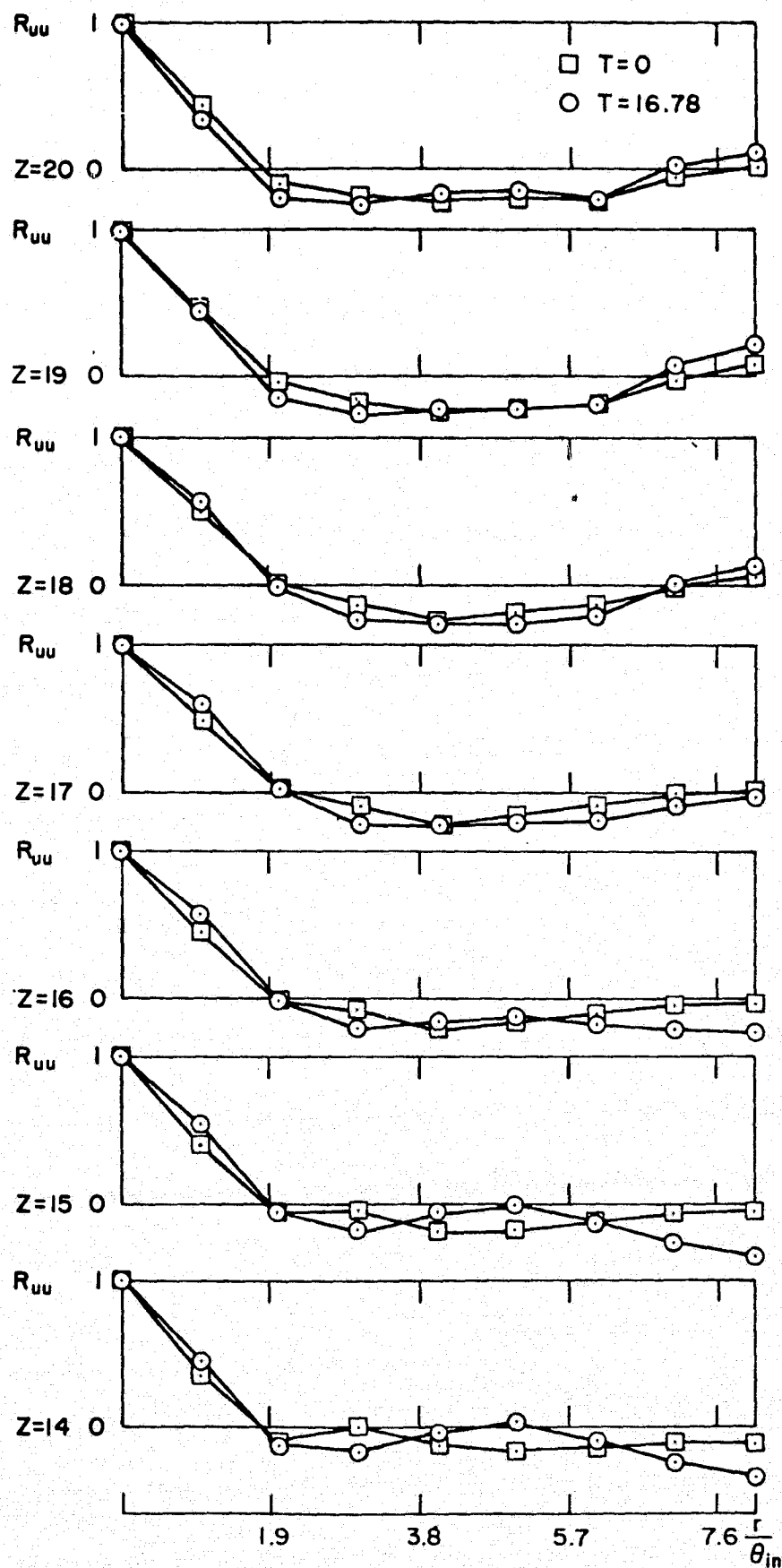


Fig. 6.11a. Two point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case a).

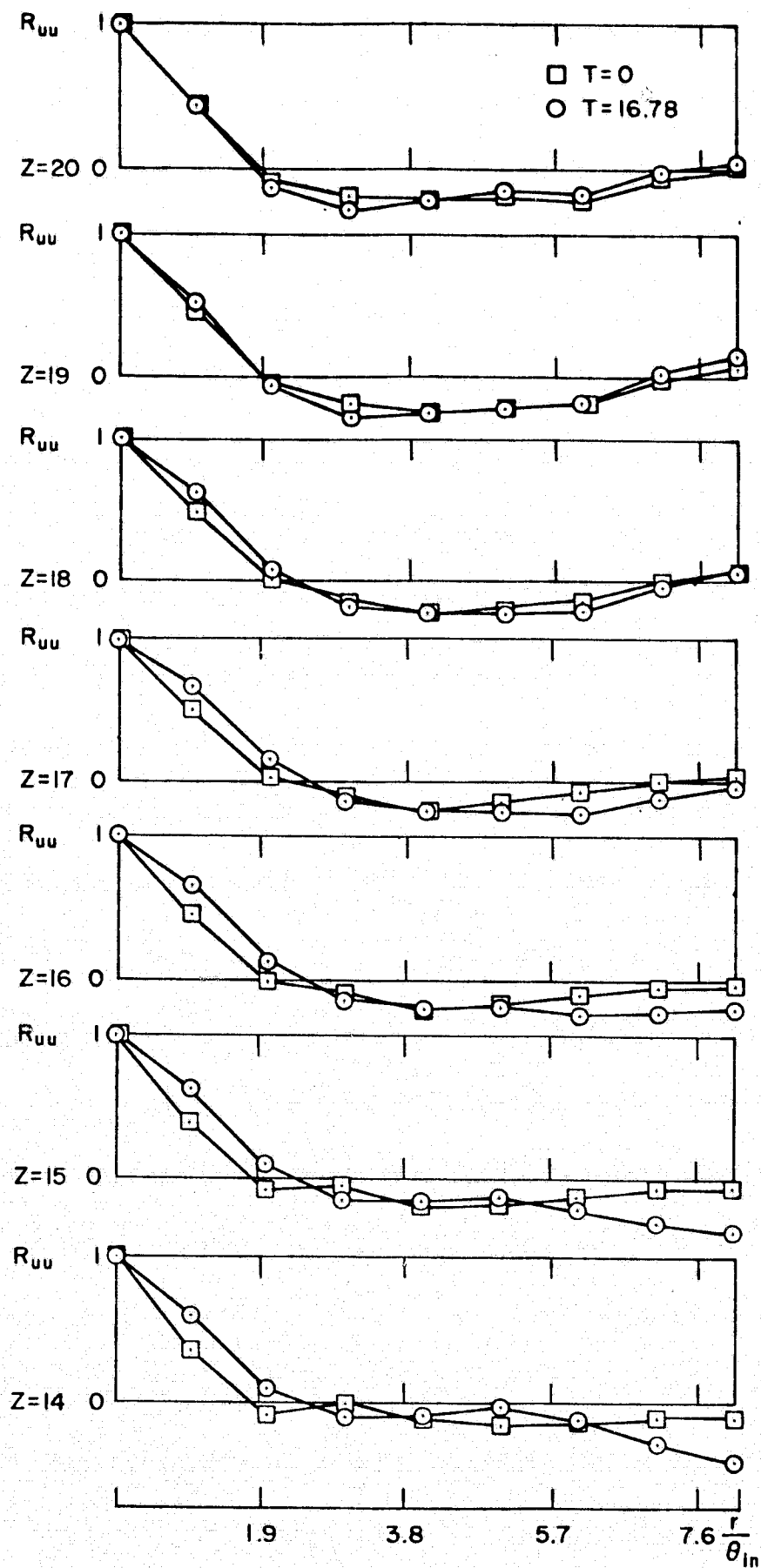


Fig. 6.11b. Two point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case b).

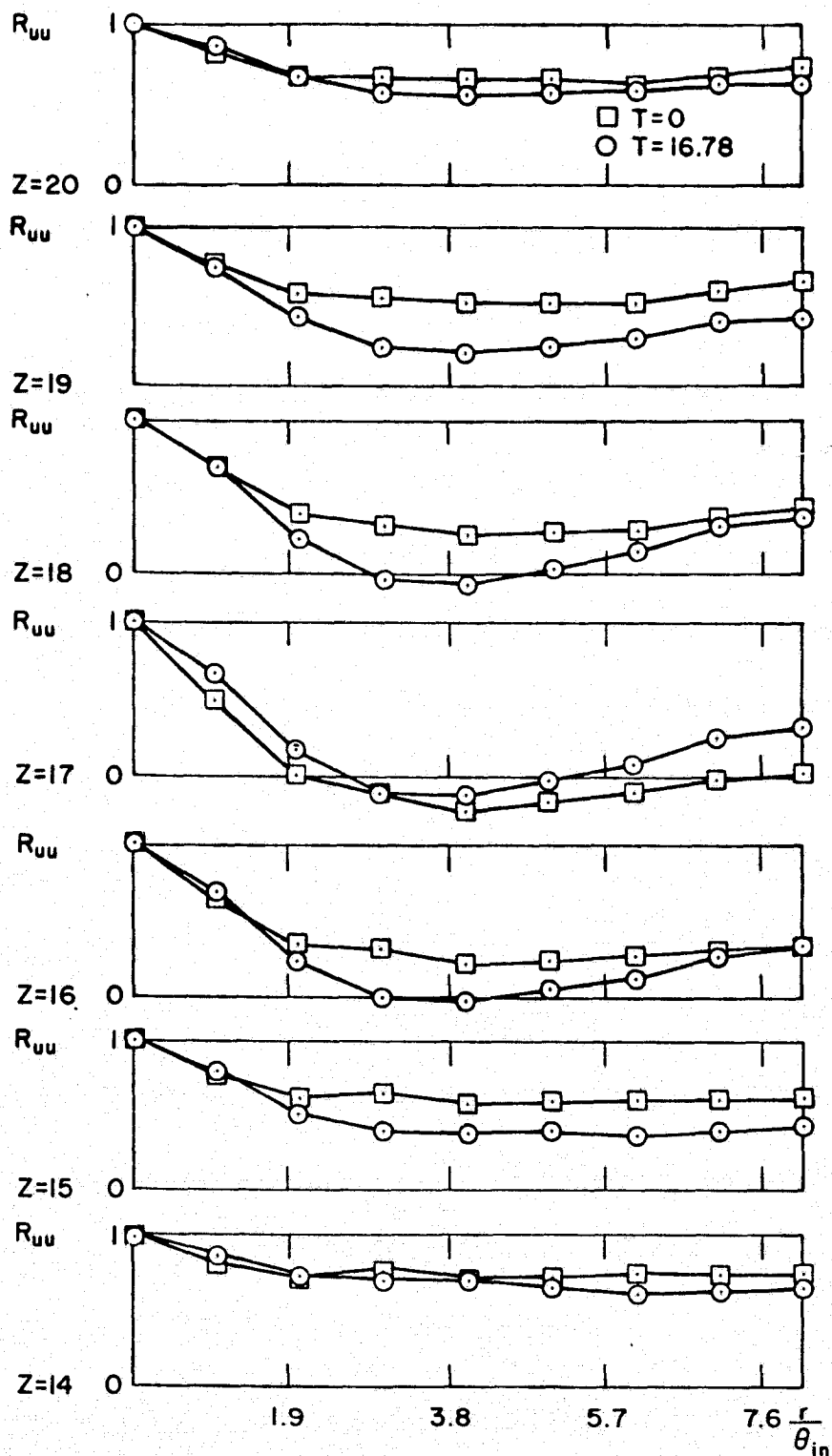


Fig. 6.11c. Two-point correlations ($z = 17$ is the center plane of the mixing layer, $\Delta z/\theta_{in} = 1.023$; case c).

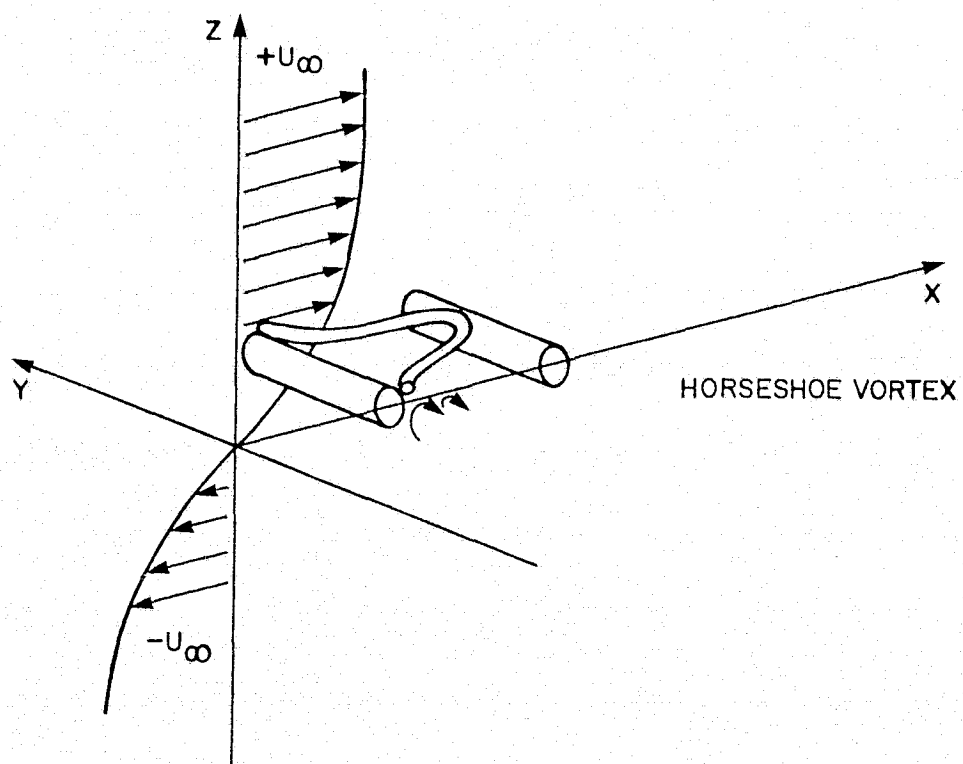


Fig. 7.1. Formation of streamwise cellular structures in a mixing layer.

Appendix A

SUBGRID SCALE MODELS FOR THE VORTICITY EQUATIONS

In Chapter 2 we propose to use the following models for W_{ij} (Eqn. (2.9)):

Model $\omega-1$

$$W_{ij} = - \epsilon_{ijk} \frac{\partial}{\partial x_k} (2v_T \bar{S}_{kl}) \quad (2.15)$$

Model $\omega-2$

$$W_{ij} = - \frac{\partial}{\partial x_j} (v_T \bar{\omega}_i) + \frac{\partial}{\partial x_i} (v_T \bar{\omega}_j) \quad (2.16)$$

where

$$v_T = (C_V \Delta)^2 (\bar{\omega}_i \bar{\omega}_i)^{1/2} \quad (2.14)$$

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial}{\partial x_j} \bar{u}_i + \frac{\partial}{\partial x_i} \bar{u}_j \right) \quad (2.12)$$

The models should satisfy the following necessary conditions:

1. they should be antisymmetric,
2. they should vanish in an irrotational region, and
3. they should be an energy sink.

Condition 1 is readily seen to be satisfied by these models. We note that in an irrotational region, $\bar{\omega}_i = 0$. Hence, $v_T = (C_V \Delta)^2 (\bar{\omega}_i \bar{\omega}_i)^{1/2} = 0$, and the model vanishes in an irrotational region; i.e., condition 2 is also satisfied.

In order to show the dissipative nature of the subgrid scale models $\omega-1$ and $\omega-2$, consider the following equation:

$$\frac{\partial \bar{\omega}_i}{\partial t} = - \frac{\partial}{\partial x_j} W_{ij} \quad (A.1)$$

where the nonlinear terms in Eqn. (2.28) have been dropped. Multiplying Eqn. (A.1) by ψ_i , and integrating over the flow volume, we get:

$$\int \psi_i \frac{\partial}{\partial t} \bar{\omega}_i dv = - \int \psi_i \frac{\partial}{\partial x_j} W_{ij} dv \quad (A.2)$$

We want to show that Eqn. (A.2) reduces to

$$\frac{\partial}{\partial t} \int \frac{1}{2} \bar{u}_i \bar{u}_i dv = - \epsilon \quad (A.3)$$

where $\epsilon \geq 0$.

Model $\omega-1$

Substituting Eqn. (2.15) in Eqn. (A.2) for W_{ij} , integrating by parts, and using periodic boundary conditions, we obtain

$$\frac{\partial}{\partial t} \int \frac{1}{2} \bar{u}_i \bar{u}_i dv = - 2 \int v_T \bar{S}_{kl} \bar{S}_{kl} dv \quad (A.4)$$

and we have for this case:

$$\epsilon = 2 \int v_T \bar{S}_{kl} \bar{S}_{kl} dv \geq 0$$

since $v_T \geq 0$.

Model $\omega-2$

In a similar way, substituting Eqn. (2.16) into Eqn. (A.2) for W_{ij} , we can show that

$$\frac{\partial}{\partial t} \int \frac{1}{2} u_i u_i dv = - \int v_T \bar{\omega}_i \bar{\omega}_i dv \quad (A.5)$$

and we have for this case

$$\epsilon = \int v_T \bar{\omega}_i \bar{\omega}_i dv \geq 0$$

Appendix B

Fast Discrete Sine Transform (FDST)

The discrete analogs to the expansion in Fourier sine series (Eqns. (3.14) and (3.15)) are

$$f(x) = \sum_{n=0}^{N-1} \hat{f}^S(n) \sin \left[\frac{n\pi x}{(N-1)h} \right] \quad (3.16)$$

$$\hat{f}^S(n) = \frac{2}{(N-1)} \sum_{j=0}^{N-1} f(x) \sin \left[\frac{n\pi x}{(N-1)h} \right] \quad (3.17)$$

where $n = 0, 1, \dots, N-1$,
 $h = L/(N-1)$,
 $x = jh, \quad j = 0, 1, \dots, N-1$,
 $N =$ number of mesh points,
 $L =$ length of the computational box.

Both the forward and inverse sine transforms involve identical sums. Eqn. (3.17) can be rewritten as:

$$\hat{f}^S(n) = -\frac{2}{(N-1)} \operatorname{Im} \left[\sum_{j=0}^{2(N-1)-1} F(x) \exp \left(\frac{-2\pi i n x}{2(N-1)h} \right) \right] \quad (B.1)$$

where

$$\begin{aligned} F(x) &= f(x) & j &= 0, 1, \dots, N-1, \\ &= 0 & j &= N, N+1, \dots, 2(N-1)-1 \end{aligned}$$

We note that the summation

$$\sum_{j=0}^{2(N-1)-1} F(x) \exp \left(\frac{-2\pi i n x}{2(N-1)h} \right) \quad (B.2)$$

is equivalent to (3.12) with $N_1 = 2(N-1)$, and an FFT routine can be used to evaluate this sum.

Fast Discrete Cosine Transform (FDCT)

The discrete analog to the expansion in Fourier cosine series (Eqns. (3.21) and (3.22)) are:

$$f(x) = \sum_{n=0}^{(N-1)} \hat{f}^c(n) \cos\left(\frac{n\pi x}{(N-1)h}\right) \quad (3.19)$$

$$\hat{f}^c(n) = \frac{2}{(N-1)} \sum_{j=0}^{(N-1)} f'(x) \cos\left(\frac{n\pi x}{(N-1)h}\right) \quad (3.20)$$

where

$$\hat{f}^c(n) = \begin{cases} \frac{1}{2} \hat{f}^c(n) & n = 0, N-1 \\ \hat{f}^c(n) & n = 1, \dots, N-2 \end{cases}$$

$$f'(x) = \begin{cases} \frac{1}{2} f(x) & j = 0, N-1 \\ f(x) & j = 1, \dots, N-2 \end{cases}$$

where

$$\begin{aligned} n &= 0, \dots, N-1 \\ h &= L/(N-1), \\ x &= jh \quad j = 0, \dots, N-1, \\ N &= \text{number of mesh points,} \\ L &= \text{length of the computational box.} \end{aligned}$$

Both the forward and inverse transforms involve identical sums.

Eqn. (3.19) can be rewritten as:

$$f(x) = \text{Re} \left[\sum_{n=0}^{2(N-1)-1} F(n) \exp\left(\frac{-2\pi i n x}{2(N-1)h}\right) \right] \quad (B.3)$$

where

$$\begin{aligned} F(n) &= \frac{1}{2} \hat{f}^c(n) & n = 0, N-1, \\ &= \hat{f}^c(n) & n = 1, \dots, N-2, \\ &= 0 & n = N, \dots, 2(N-1)-1. \end{aligned}$$

We note that the sum in (B.3) is identical to the sum (B.2), and an FFT routine can be used to evaluate it. In fact, the sine and cosine transforms can be done simultaneously, if it is necessary to have both.

Appendix C

Effect of a Sinusoidal Vorticity Perturbation on a Uniform Vortex Array

In Chapter 5 we have studied the effect of perturbing a uniform array of vortices by offsetting the spacing of the vortices ($\beta > 0$). In this appendix we study the effect of adding a sinusoidal vorticity perturbation to a uniform array of vortices ($\beta = 0$).

1. Initial Conditions

The initial conditions studied in this appendix were generated by starting with a uniform array of point vortices on the centerline of our computational box:

$$\omega_{2u} = C_1 \left(\delta(x-L_1/4) + \delta(x-3L_1/4) \right) \delta(z-L_3/2) \quad (C.1)$$

We then add a cosine vorticity distribution to (C.1):

$$\omega_2 = \omega_{2u} - C_2 \cos \left(\frac{2\pi x}{L} \right) \delta(z-L_3/2) \quad (C.2)$$

Eqn. (C.2) is then filtered with a relatively wide Gaussian filter (Eqn. (5.19)) to yield the initial conditions. The initial velocity is then non-dimensionalized on Δu and the length scales on θ_{in} . The computational details, i.e., number of mesh points, mesh size, time steps, and boundary conditions, are the same as in Chapter 5. Only the initial conditions were changed.

2. Results

The momentum thickness (θ) is defined by Eqn. (5.4). Fig. C.1 shows θ/θ_{in} plotted vs. T for $C_2/C_1 = 0.1/20, 1/20, 2/20, 4/20$. We note that the growth rate of the layer is highly dependent on the strength of the perturbation. The growth rate more than doubles from 0.016 to 0.035 when the strength of the perturbation is doubled (C_2/C_1 from 2/20 to 4/20).

We note also that for high amplitude perturbations, $C_2/C_1 = 4/20$, the growth rate starts to level off for $T > 12.0$. This saturation is also observed experimentally by Oster et al. (1978); they have oscillated the initial conditions of a two-dimensional mixing layer.

Figures C.2 and C.3 show the non-dimensional mean velocity and turbulence intensity (as in Sections 5.7 and 5.8) plotted vs. z/θ for $C_2/C_1 = 2/20$. We note that the mean velocity profiles are self-similar. This is not surprising, since self-similarity of the mean velocity profiles is easily obtained (see Section 6.5). Turbulence intensity profiles (Fig. C.3) show that self-similarity is also more or less obtained for the present case.

These results are similar to those obtained in Chapter 5 by using a spacing perturbation. Apparently the perturbation can take any of a number of forms, and the characteristics of the shear layer will be nearly the same. Under experimental conditions, the nature of the perturbation is difficult to determine. What we do note is that reproduction of the experimentally observed growth rate does require large perturbations, which are apparently created by either the inflow or outflow conditions of the experiment.

Appendix D

INTERACTIONS BETWEEN STREAMWISE AND SPANWISE VORTICITY

In Chapter 6 we studied the effect of a random fluctuation on vortex pairing. In this appendix we study the interactions between a streamwise cellular vortex structure and spanwise vortex pairing.

1. Initial Conditions

The initial conditions studied in this appendix were generated by adding to a row of spanwise vortices ($\beta = 3/16$) a row of streamwise vortices of alternating signs:

$$\bar{\omega}_1 = C_2 \sin\left(\frac{2\pi y}{L_2}\right) \exp\left(-\frac{(z-L_3/2)^2}{6h_3^2}\right) \quad (D.1)$$

The same computational setup described in Chapter 6 is used, i.e., the same boundary conditions, number of mesh points, mesh sizes, and time step.

Figure D.1 shows a contour map in the y - z plane of the streamwise vorticity. We note that $\bar{\omega}_1$ displays a cellular structure and that $\bar{\omega}_1$ does not initially have a streamwise variation. We ran two cases:

Case a:

$$\frac{|\omega_1|_{\max}}{|\omega_2|_{\max}} = 0.037$$

Case b:

$$\frac{|\omega_1|_{\max}}{|\omega_2|_{\max}} = 0.370$$

2. Results

We first look at the development of the momentum thickness, $\theta(t)$, defined by Eqn. (5.4), in time. The non-dimensional mean velocity (Section 5.7) and mean turbulence intensity (Section 5.8) are also considered. The interaction between the spanwise vortices and the streamwise vortices is studied using contour plots. Note that we have a three-dimensional box and that contour plots in different planes for different vorticity directions will be considered.

Figure D.2 shows θ/θ_{in} plotted vs. T . The momentum thickness growth rate, $d\theta/\Delta u dt = 0.020$, for Case (a) is the same as it was in the absence of the streamwise vortices. However, the momentum thickness growth rate, $d\theta/\Delta u dt = 0.040$, doubled for Case (b).

Figures D.3a and -b show $2\langle u \rangle_{xy}/\Delta u$ plotted vs. z/θ for Cases (a) and (b), respectively, at $\Delta T = 2.4$ intervals. We note that both cases produce self-similar mean velocity profiles.

Figures D.4a and -b show $q^2/2(\Delta u)^2$ plotted vs. z/θ for Cases (a) and (b), respectively, at $\Delta T = 2.4$ intervals. The mean turbulence intensity results for Case (a) are similar to those we obtained when the streamwise vortices were not present. As in the 2-D case (with $\beta = 3/16$), the mean turbulence intensity decays slightly, then reaches a self-similar situation. For Case (b), in which we have strong streamwise vortices, Fig. D.4b shows that the turbulence intensity grows with time, and the profiles do not show self-similarity.

(a) Contour Plots in the x - z Planes

Figures D.5 show constant vorticity contours of the spanwise ($\bar{\omega}_2$) vorticity at time $T = 16.78$. In both cases the spanwise vortices have paired. The shapes are similar, but the roller is slightly distorted for Case (b) as compared to Case (a) and to the 2-D results (see Fig. 5.7d). This indicates that the streamwise vortices did not affect the merging of the spanwise vortices, but the strong streamwise vortices (Case (b)) have affected the shape of the roller.

Figures D.6 show constant vorticity contours of the streamwise vorticity for Cases (a) and (b). These figures indicate that the streamwise vortices have been convected to the edges of the mixing layer by the spanwise vortices. There is also clear evidence of vortex stretching.

Figure D.7 shows the projection of the vorticity vector at $T = 16.78$, for Case (b). We can see clearly that the originally straight vortex lines have been convected and stretched by the spanwise vortices to assume an inverted S shape.

(b) Contour Plots in the y-z Planes

Figure D.8 shows constant vorticity contours of the spanwise vorticity for Case (b). The spanwise vortices have been convected and stretched by the strong counter-rotating streamwise vortices and exhibit spanwise waviness. This means that the contact area between the rotational fluid and the irrotational fluid has increased, which leads to an increase in the entrainment rate. This waviness also explains the increase in the turbulence intensity and high growth of the momentum thickness of the mixing layer. Note that the mean quantities are defined as horizontal planar averages and, with this definition, the wavy layer appears thicker and more turbulent than a strictly two-dimensional layer.

The above results indicate that the effect of the streamwise vorticity on the spanwise vorticity is almost independent of the effect of the spanwise vorticity on the streamwise vorticity. Indeed, a straight line of particles placed at the center of the layer in the streamwise direction would be convected to form an inverted S shape in the presence of the two-dimensional vortex pairing. A straight line of particles initially passing through the center of an array of counter-rotating vortices will be convected to assume a wavy shape.

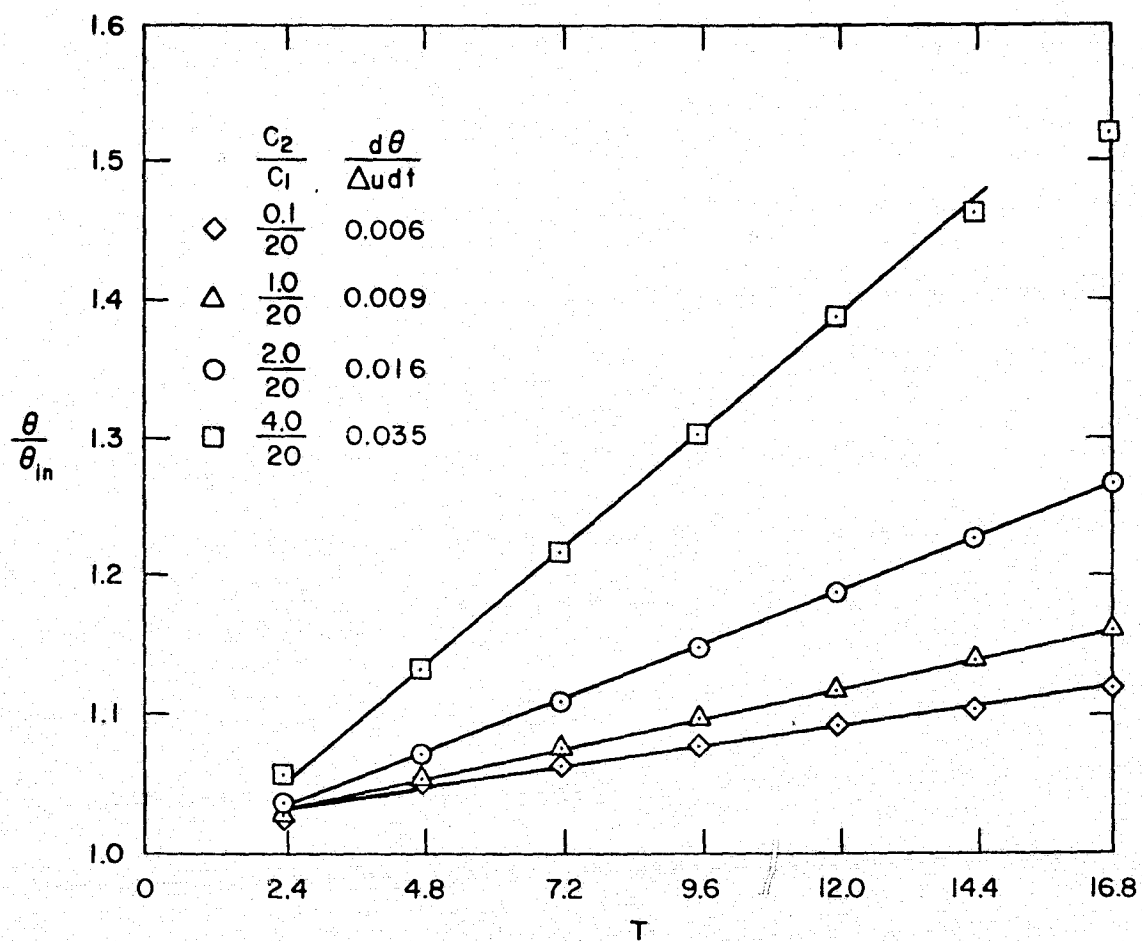


Fig. C.1. Non-dimensional momentum thickness (θ/θ_{in}) as a function of time for various C_2/C_1 .

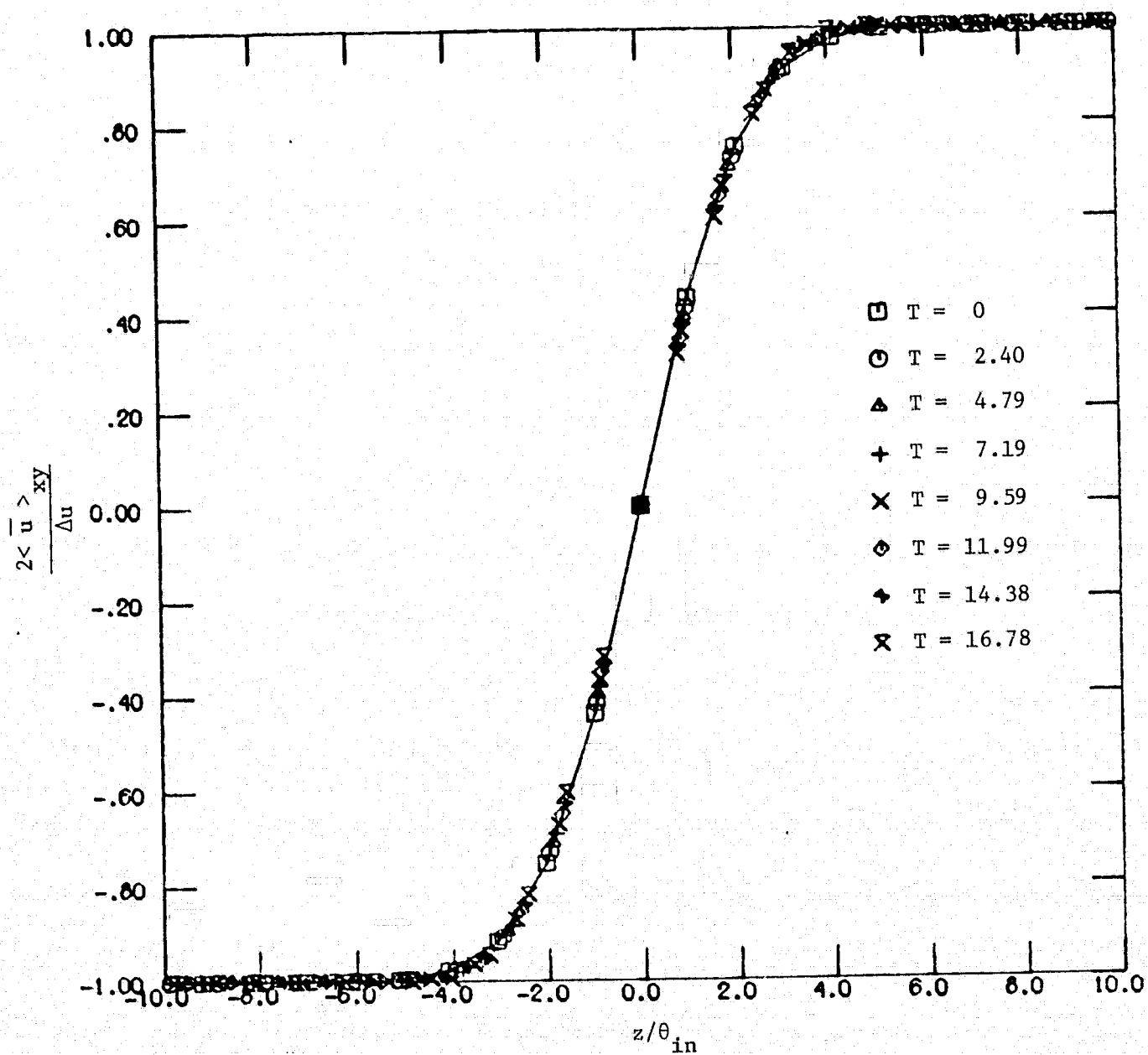


Fig. C.2. Mean velocity profiles ($C_2/C_1 = 0.1$)

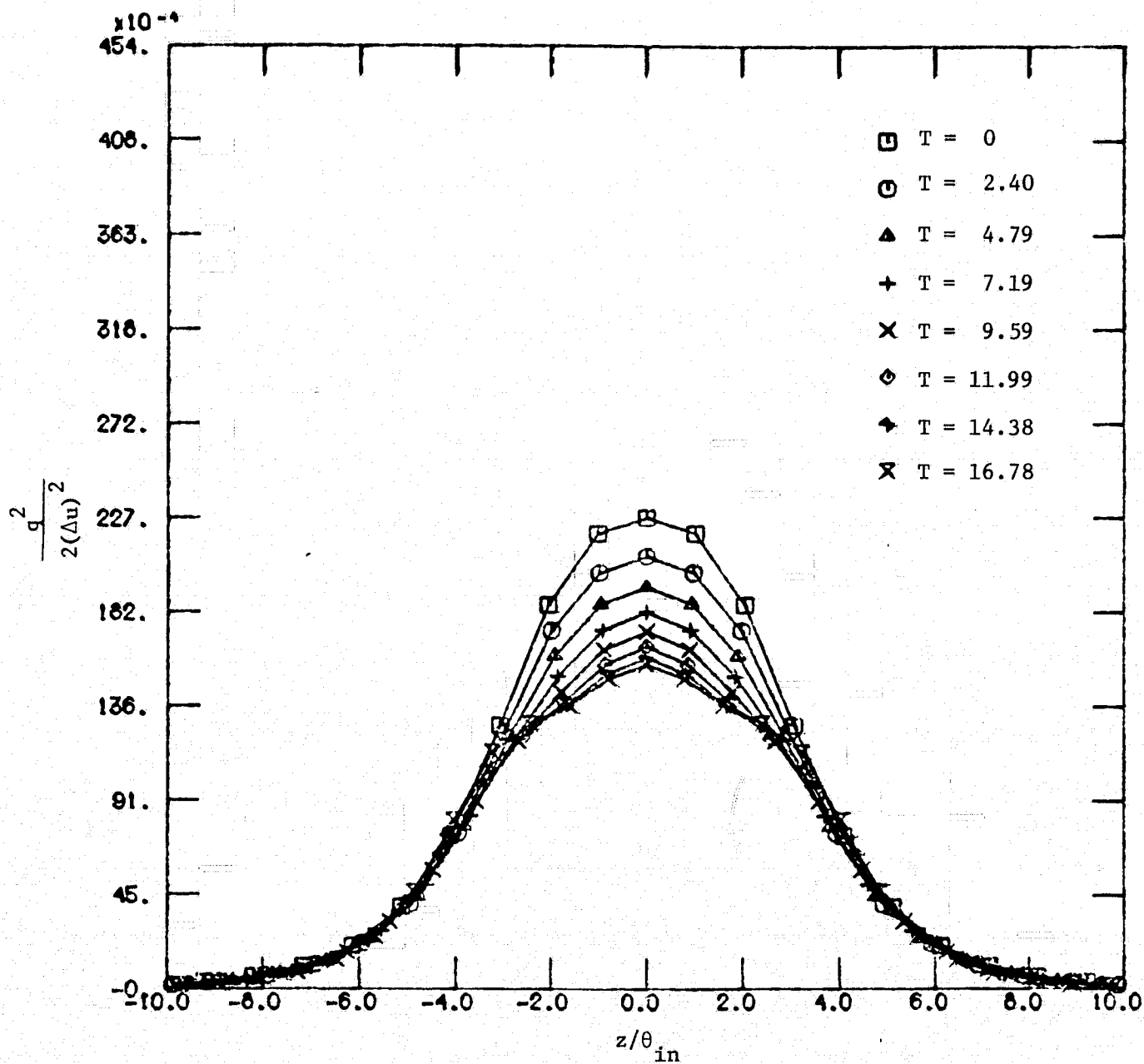


Fig. C.3. Mean turbulence intensity profiles ($C_2/C_1 = 0.1$).

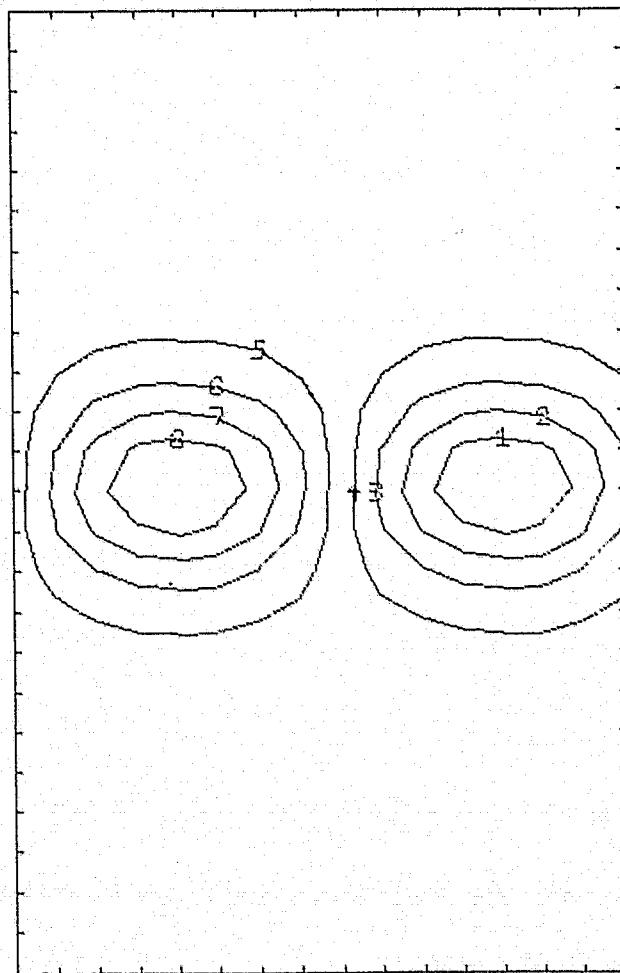


Fig. D.1. Contour plots of the streamwise vorticity ($\bar{\omega}_1$) at time $T = 0$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels.

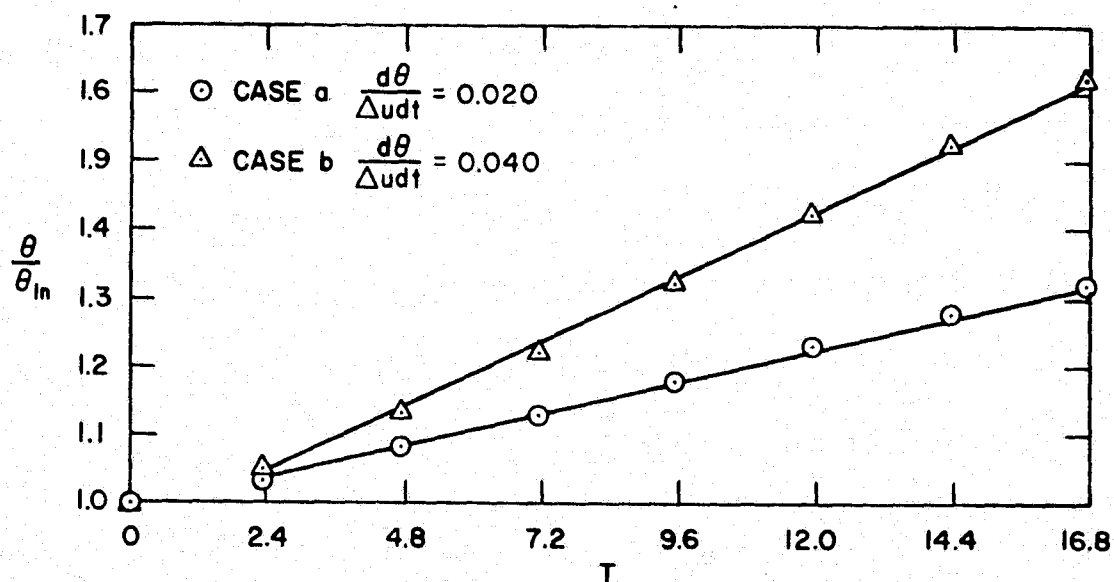


Fig. D.2. Non-dimensional momentum thickness (θ/θ_{in}) as a function of time.

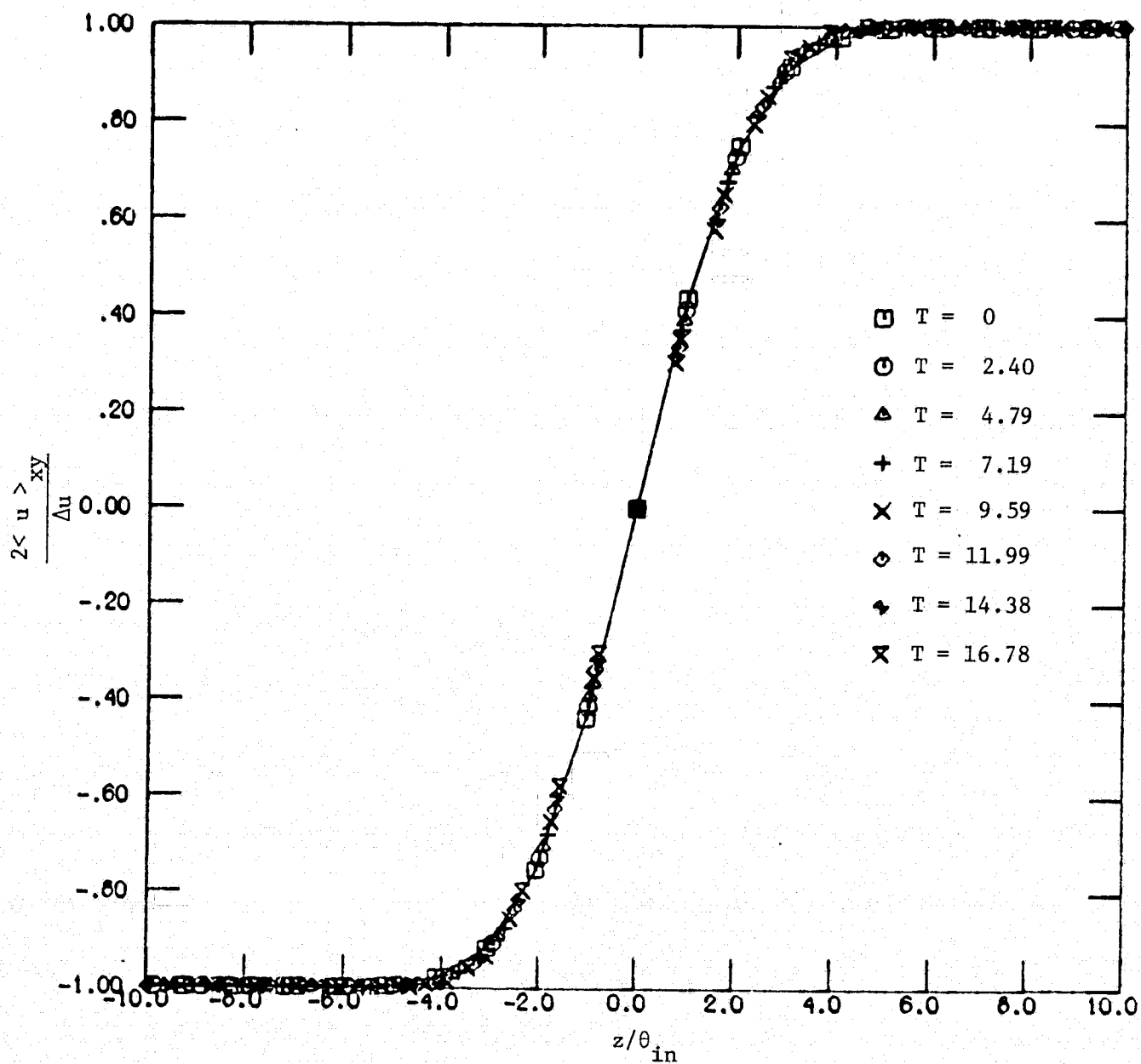


Fig. D.3a. Mean velocity profiles (case a)

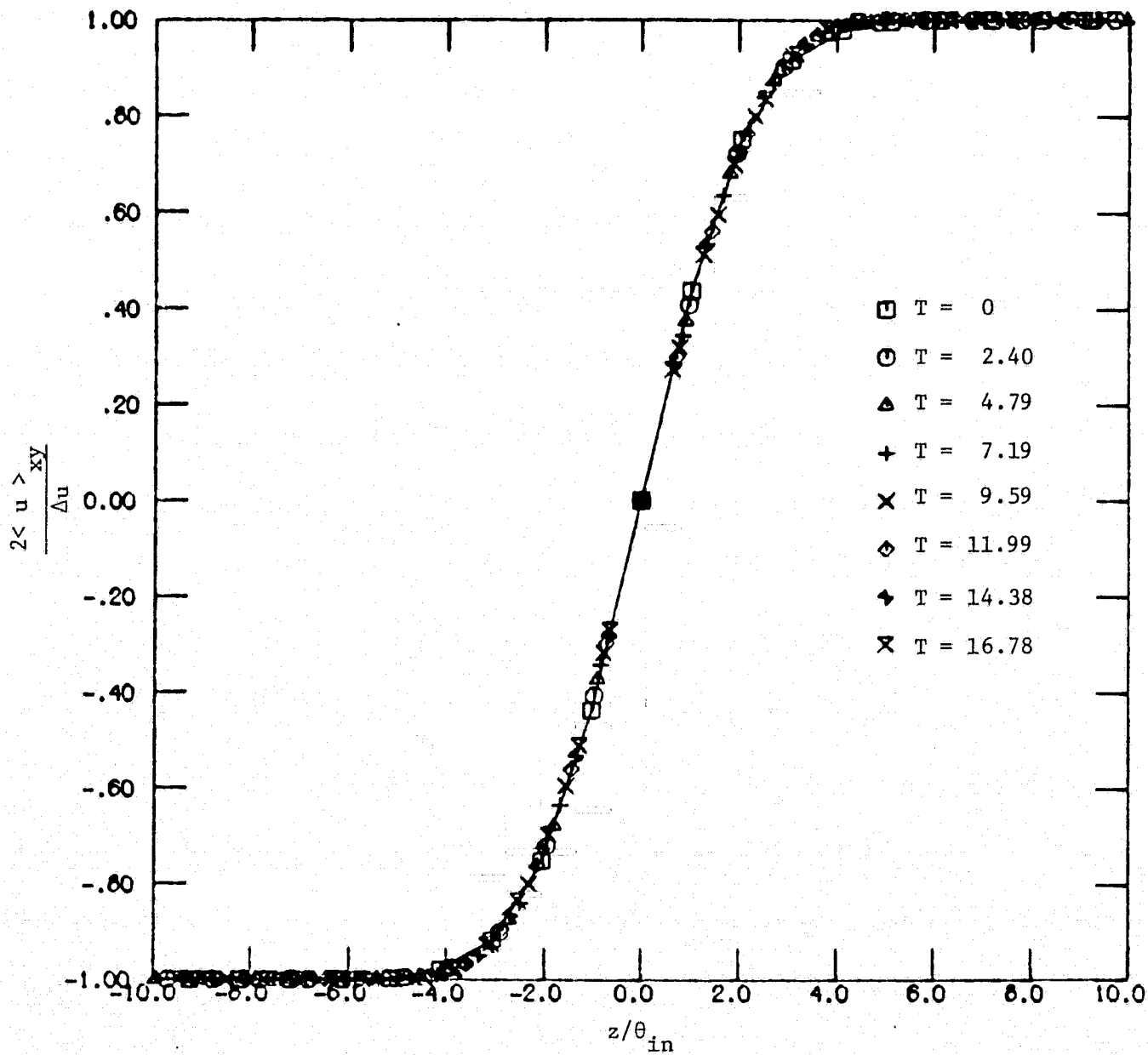


Fig. D.3b. Mean velocity profiles (case b)

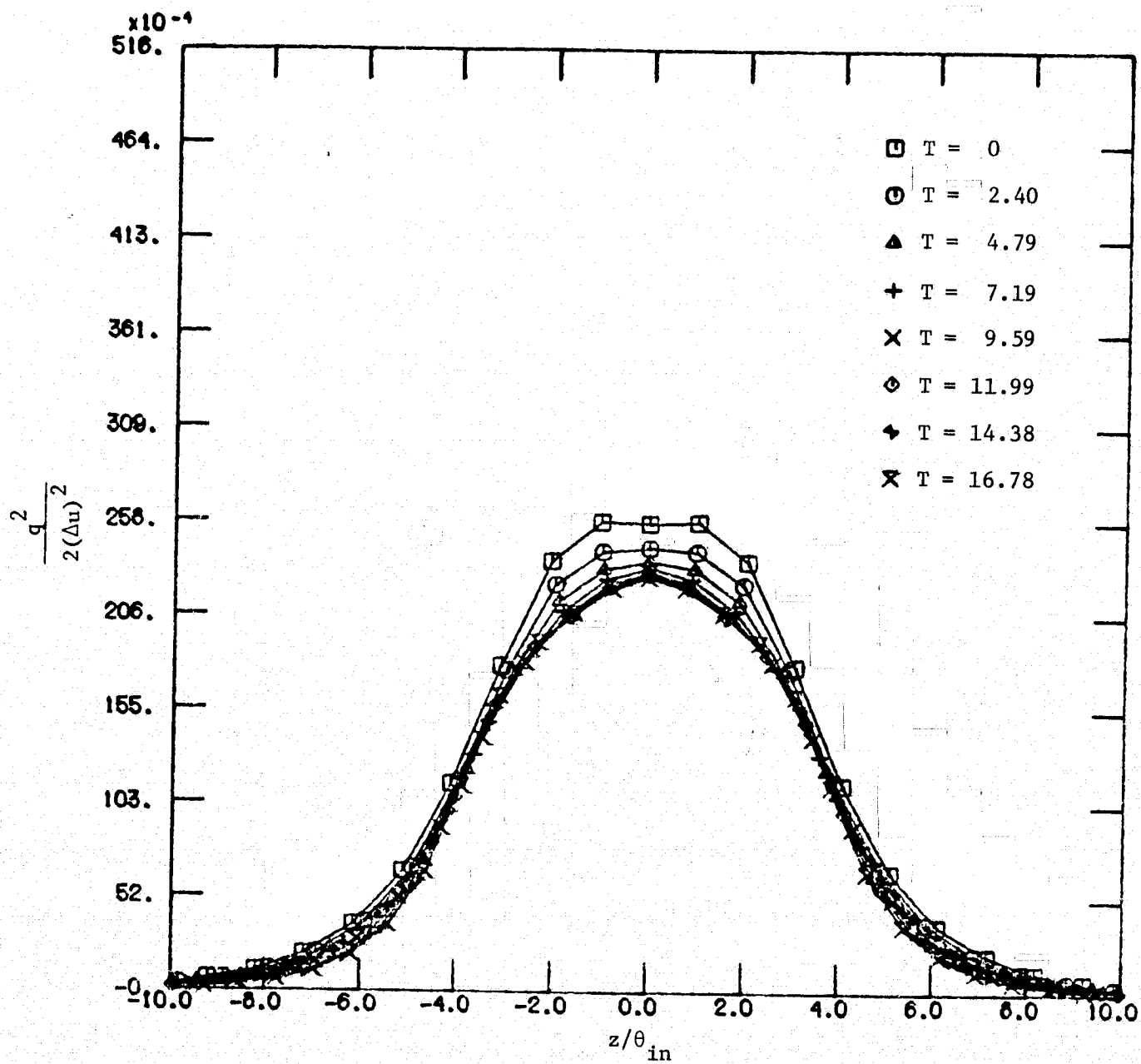


Fig. D.4a. Mean turbulence intensity profiles (case a).

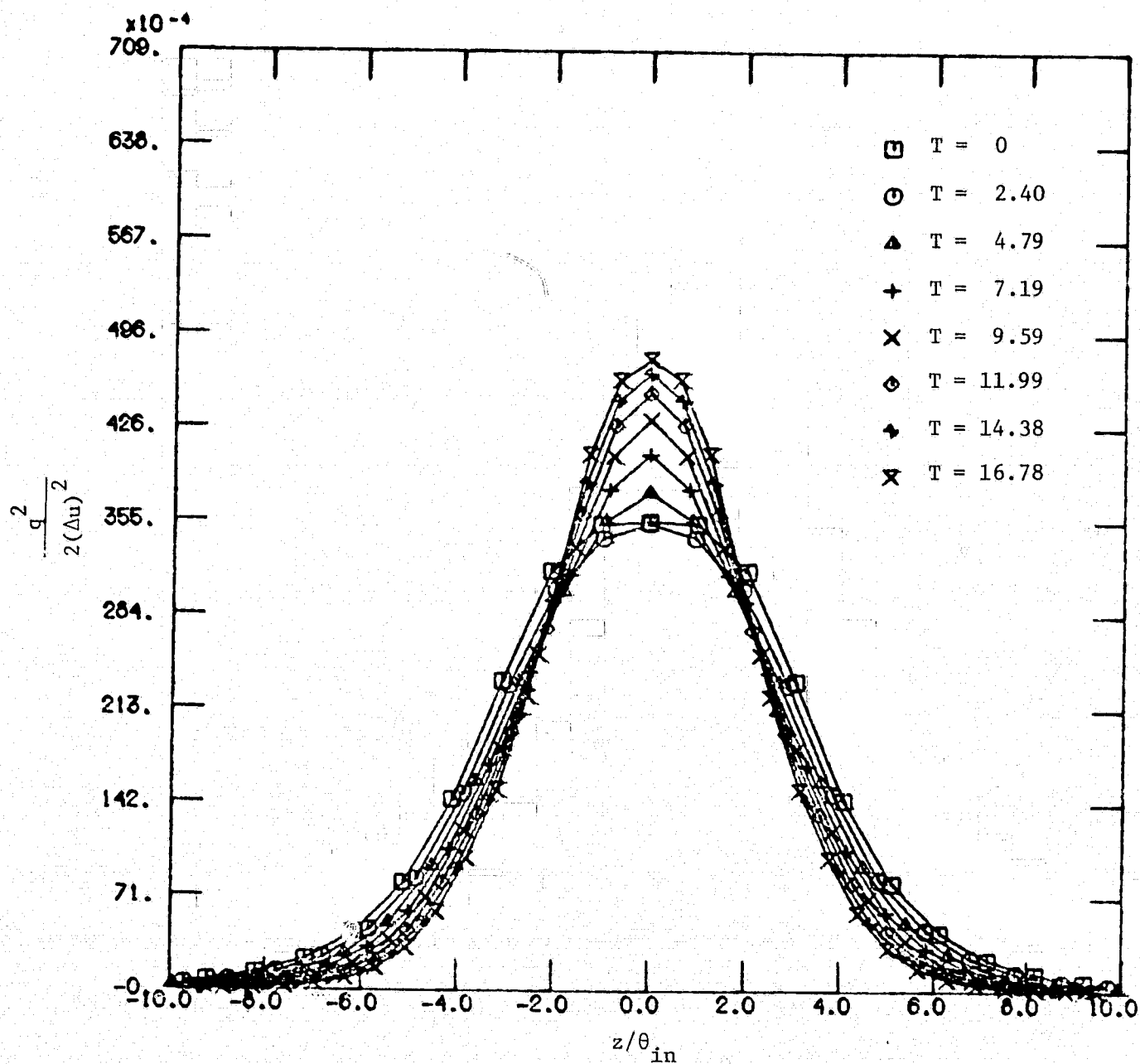


Fig. D.4b. Mean turbulence intensity profiles (case b).

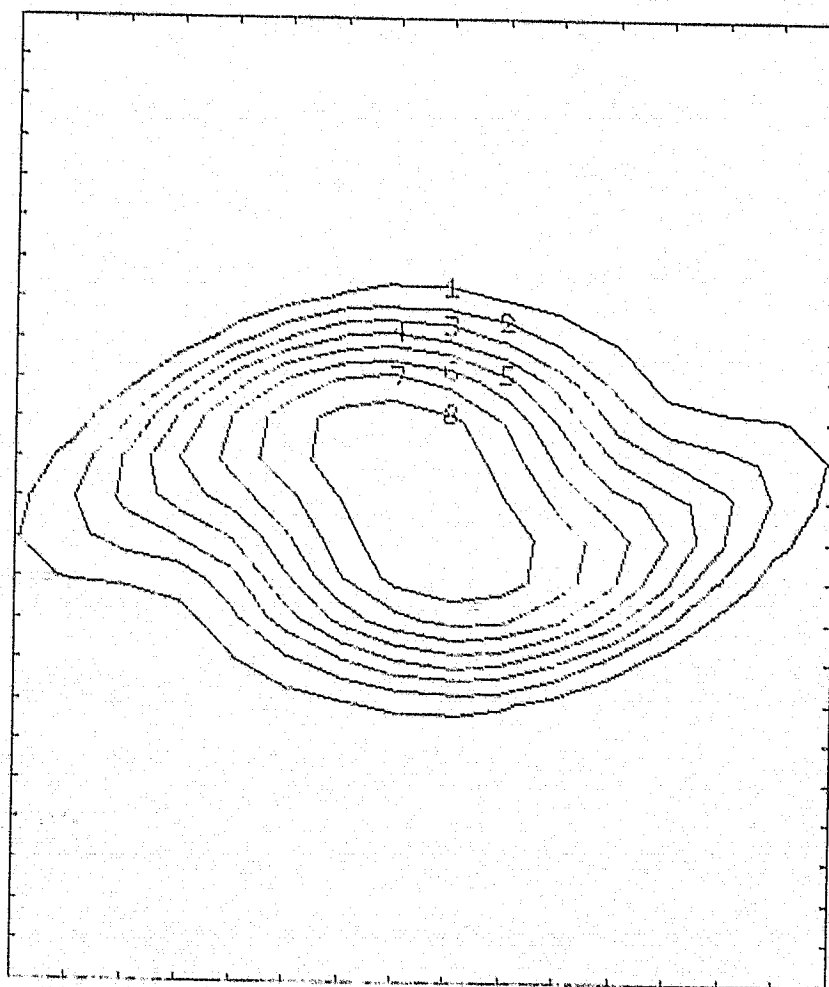


Fig. D.5a. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case a).

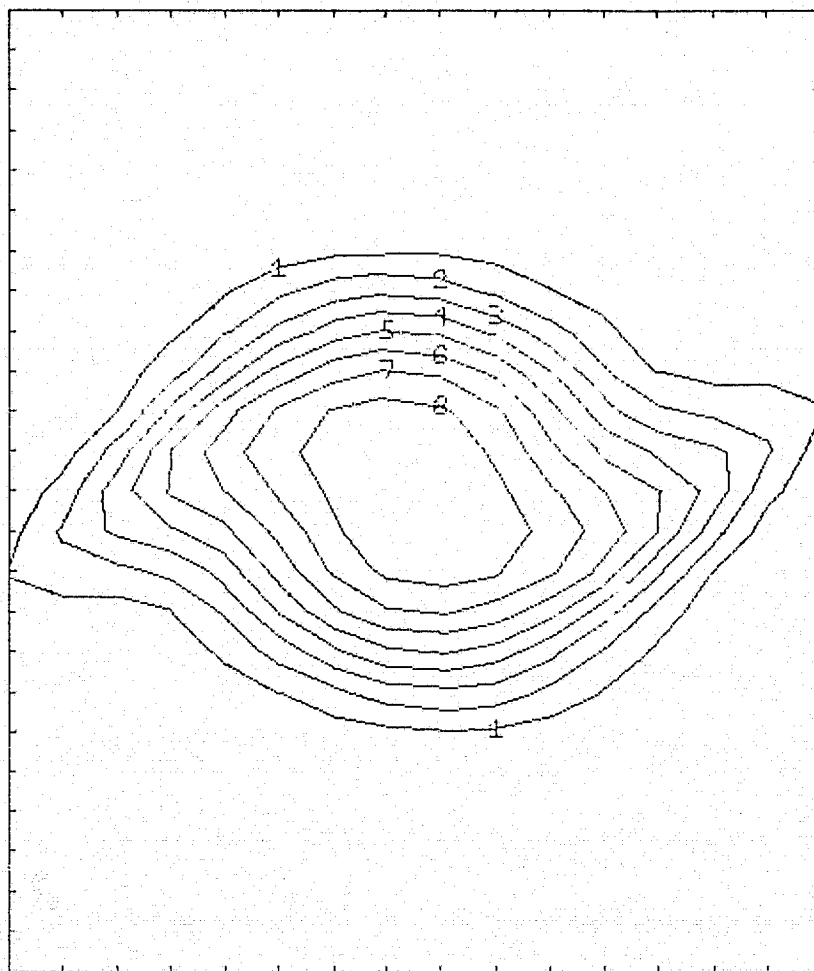


Fig. D.5b. Contour plots of the spanwise vorticity ($\bar{\omega}_2$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case b).

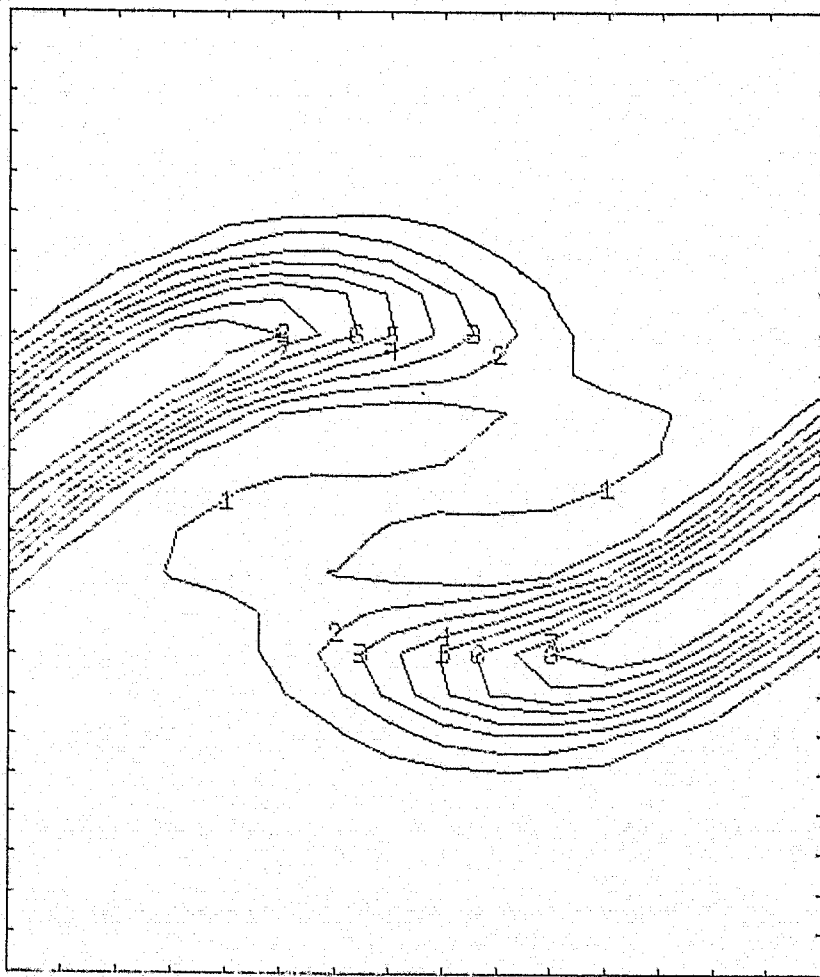


Fig. D.6a. Contour plots of the streamwise vorticity ($\bar{\omega}_1$) in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels (case a).

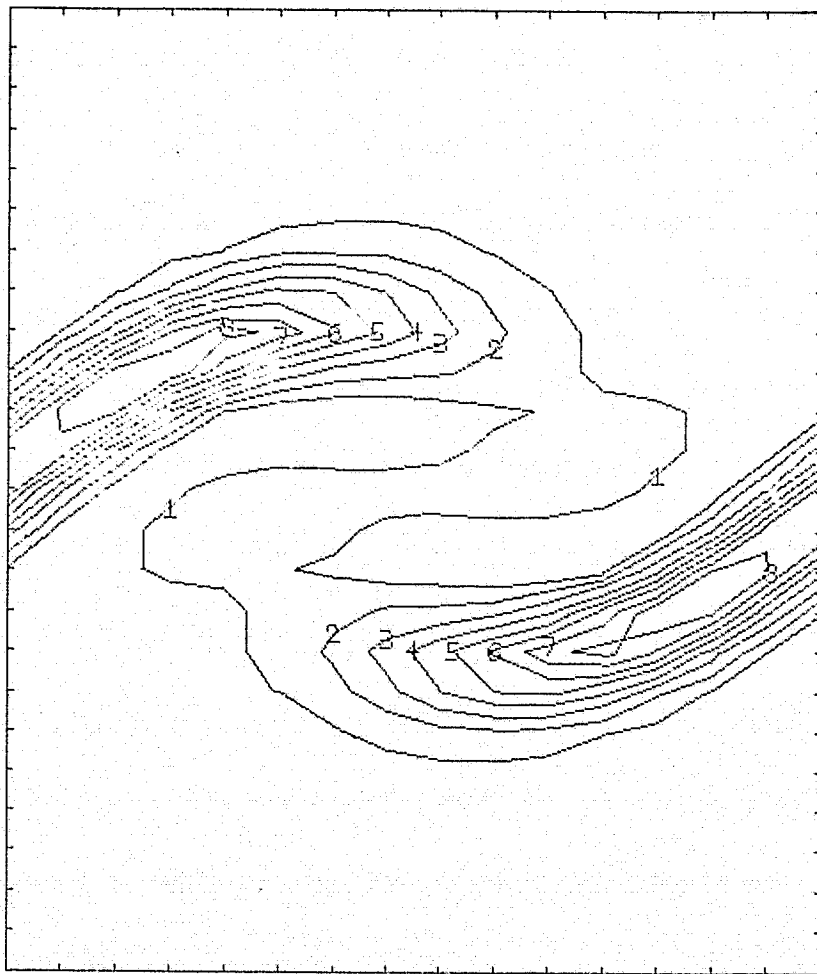


Fig. D.6b. Contour plots of the streamwise vorticity $(\bar{\omega}_1)$ in an x-z plane ($y/\theta_{in} = 4.09$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher levels (case b).

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

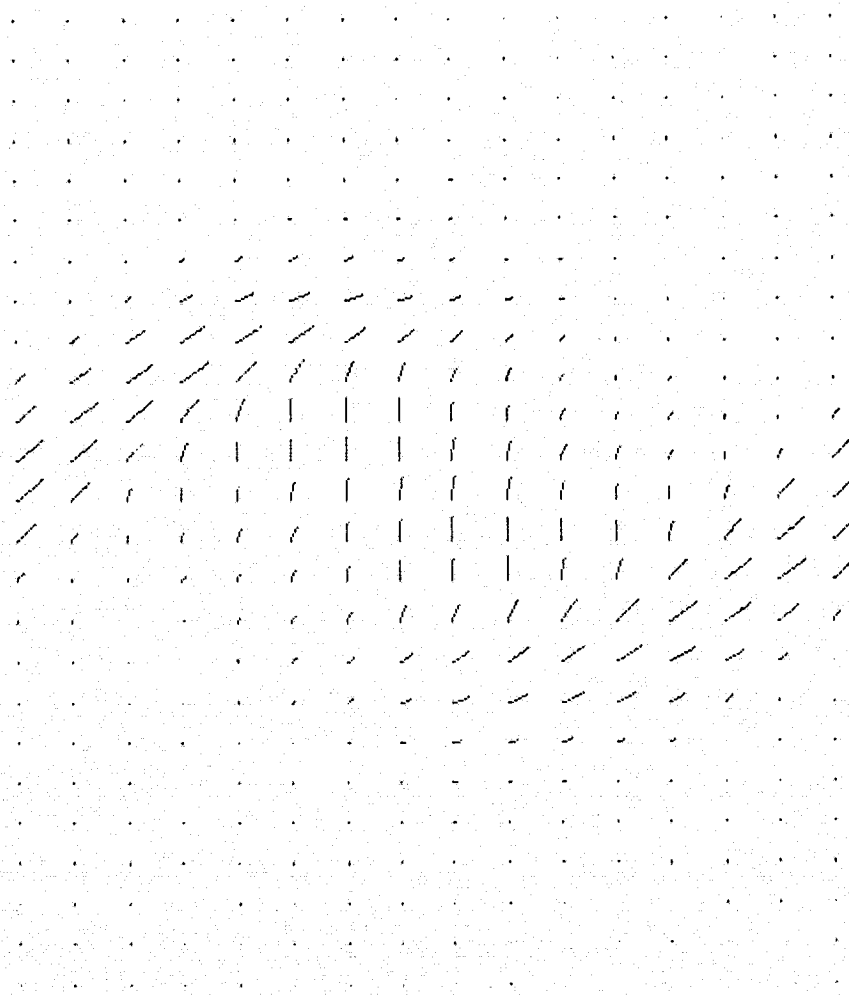


Fig. D.7. Projection of the vorticity vector in an $x-z$ plane
($y/\theta_{in} = 4.09$) at time $T = 16.78$.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

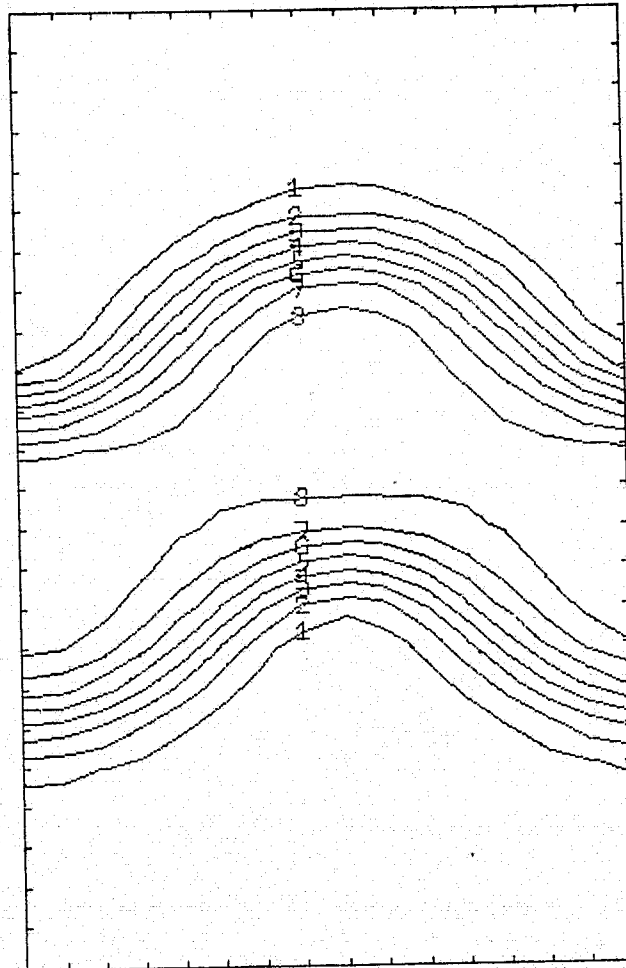


Fig. D.8. Contour plots of the spanwise vorticity. $(\bar{\omega}_2)$ in a y - z plane ($x/\theta_{in} = 9.55$), at time $T = 16.78$. Constant vorticity lines are plotted at eight levels. Higher numbers on these lines indicate higher vorticity levels ($\bar{\omega}_{2,max} = 0.714$ case b).

C-----COMPUTER PROGRAM WRITTEN TO CALCULATE TURBULENT MIXING LAYERS

C
C

*COMDECK AVG

COMMON/AVG/ AVG1,AVG2,AVG3,CCF

*COMDECK BLANK

COMMON/DUDX(16,16,33)

*COMDECK DATA7

COMMON/DATA7/ FR(16,16),FI(16,16)

*COMDECK DATA9

COMMON/DATA9/ IMAX,JMAX,LMAX

*COMDECK DAT21

COMMON/DAT21/ XR(64),XI(64)

*COMDECK DEL

COMMON/DEL/ DELTAX,DELTAY,DELTAZ

*COMDECK DIM

COMMON/DIM/N1,N2,N3

*COMDECK FLT

COMMON/FLT/ FILT1(16),FILT2(16),FILT3(33)

*COMDECK LARGE2

COMMON/LARGE2/ U(16,16,33),V(16,16,33),W(16,16,33)

LEVEL 2,U,V,W

*COMDECK LARGE3

COMMON/LARGE3/ GU(16,16,33),GV(16,16,33),GW(16,16,33)

LEVEL 2,GU,GV,GW

*COMDECK LARGE5

COMMON/LARGE5/ 01(16,16,33),02(16,16,33),03(16,16,33)

LEVEL 2,01,02,03

*COMDECK MEANVOR

COMMON/MEANVOR/ VOR(32,33)

*COMDECK PR

COMMON/PR/ CCPW,CCPF,CCPD

*COMDECK WV

COMMON/WV/ WAVEX(16),WAVEY(16),WAVEZ(33),WAVEXS(16),WAVEYS(16)

1 ,WAVEZS(33)

*COMDECK XL

COMMON/XL/ XPART(160),YPART(160),ZPART(160),NCHAR(160)

*DECK MAIN

PROGRAM MAIN(INPUT,OUTPUT,TAPE8,TAPE9,TAPE10)

C*****

C MAIN CONTROLS THE COMPUTATION SEQUENCES. *

C IN THIS ROUTINE WE ADVANCE IN TIME. *

C THE EXTERNALS USED IN THIS ROUTINE ARE *

C CFILTER *

C CONVEC *

C CURLO *

C DATARED *

C EDVIS *

C INVERS *

C MEANINI *

C MOVLEV *

C SFILTER *

C SFILTER *

C SGS *

C STFILT *

C STREAD *

C STWV *

C*****

INTEGER TIME,TSTART,TEND

COMMON/TIM/ TSTART,TEND

COMMON/LARGE4/ RU(16,16,33),RV(16,16,33),RW(16,16,33)

COMMON/NORM/ DELU,THETA

LEVEL 2,RU,RV,RW

COMMON/DATCNT/ IDATCNT

*CALL MEANVOR

*CALL XL

COMMON/CONST/C100,C101,IJK,IJ,NHP1,HALF

*CALL DAT21

*CALL LARGE2

*CALL BLANK

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

*CALL LARGE5
*CALL DEL
*CALL DATA9
*CALL WV
*CALL DATA7
*CALL FLT
*CALL LARGE3
*CALL DIM
*CALL AVG
C   START THE READOUT OF INPUT
    CALL STREAD
C   SET THE COEFFICIENT OF THE SUBGRID SCALE MODEL
    C=0.188
C   SET COF = 1 FOR THE FIRST TIME STEP
    COF=1.0
    IJ=N1*N2
    IJK=N1*N2*N3
    DO 1 L=1,LMAX
    DO 1 J=1,JMAX
    DO 1 I=1,IMAX
    U(I,J,L)=0.
    V(I,J,L)=0.
    W(I,J,L)=0.
    O1(I,J,L)=0.
    O2(I,J,L)=0.
    O3(I,J,L)=0.
1   CONTINUE
C   SET THE WAVE NUMBERS
    CALL STWV
C*****SET THE INITIAL CONDITIONS
    CALL MEANINI
C   NON DIMENSIONALIZE THE TIME STEP ON DELU/THETA
    DT=0.03125*DELU/THETA
C   SET THE FILTER WIDTH = 2*MESH SIZE
    AVG1=2.0
    AVG2=2.0
    AVG3=2.0
    COEF2=(CX*(AVG1*DELTAX*AVG2*DELTAY*AVG3*DELTAZ)**(1./3.))**2
    DO 123 L=1,LMAX
    DO 123 J=1,JMAX
    DO 123 I=1,IMAX
    RU(I,J,L)=0.
    RV(I,J,L)=0.
    RW(I,J,L)=0.
123  CONTINUE
    ICOUNT=0
    TIME=0
C   WRITE ON TAPE 9 TO BE STORED ON DISC PACK
    PRINT 1100,TIME
    WRITE(9) TIME,O1,O2,O3,DT,DELTAX,DELTAY,DELTAZ,DELU,THETA
    IDATCNT=0
C   COMPUTE THE STATISTICS OF THE INITIAL CONDITIONS
    CALL DATARED
C   SET THE FOURIER TRANSFORM OF THE GAUSSIAN FILTER
    CALL STFILT
    IDJM=30
    DO 300 TIME=ISTART,TEND
C*****COMPUTE THE ADVECTIVE AND STRETCHING TERMS
    CALL CONVEC
    CALL SFILTER(GU,DUDX,N1,N2,N3)
    CALL SFILTER(GV,DUDX,N1,N2,N3)
    CALL CFILTER(GW,DUDX,N1,N2,N3)
C*****COMPUTE THE EDDY VISCOSITY
    CALL EDVIS(COEF2,DUDX,N1,N2,N3)
C*****COMPUTE THE SGS MODEL
    CALL SGS(U,V,W,N1,N2,N3)
C*****ADVANCE IN TIME
    DO 800 L=1,LMAX
    DO 800 J=1,JMAX
    DO 800 I=1,IMAX

```

```

      01(I,J,L)=01(I,J,L)+DT*(COF*GU(I,J,L)-0.5*RU(I,J,L))
      02(I,J,L)=02(I,J,L)+DT*(COF*GV(I,J,L)-0.5*RV(I,J,L))
      03(I,J,L)=03(I,J,L)+DT*(COF*GW(I,J,L)-0.5*RW(I,J,L))
      800 CONTINUE
C*****STORE THE PREVIOUS TIME STEP
      CALL MOVLEV(GU(1,1,1),RU(1,1,1),IJK)
      CALL MOVLEV(GV(1,1,1),RV(1,1,1),IJK)
      CALL MOVLEV(GW(1,1,1),RW(1,1,1),IJK)
C*****THE VORTICITY AT THE NEXT TIME STEP HAS BEEN COMPUTED
C*****FIND THE CORRESPONDING VELOCITY FIELD
      CALL INVERS(01,GU,DUDX,1,N1,N2,N3)
      CALL INVERS(02,GV,DUDX,2,N1,N2,N3)
      CALL INVERS(03,GW,DUDX,3,N1,N2,N3)
      CALL CURLO(GU,GV,GW,U,V,W,N1,N2,N3)
C      SET COF = 1.5 FOR SUBSEQUENT TIMES (ADAMS-BASHFORTH)
      COF=1.5
      ICOUNT=ICOUNT+1
      IICOUNT=ICOUNT-IDUM
      IF (IICOUNT .NE. 0) GO TO 300
      ICOUNT=0
      PRINT 1100,TIME
      WRITE(9) TIME,01,02,03
      CALL DATED
      300 CONTINUE
      1100 FORMAT(1H1,5X,* TIME STEP =*,I5)
      1000 FORMAT(1P8E15.7)
      STOP
      END
*DECK CFILTER
      SUBROUTINE CFILTER(HR,HI,N1,N2,N3)
C*****
C      CFILTER COMPUTES THE FILTER OF THE HR VARIABLE BY EXPANDING IN
C      FOURIER SERIES IN THE X- AND Y- DIRECTIONS AND FOURIER COSINE
C      SERIES IN THE Z-DIRECTION
C      THIS ROUTINE USES AS EXTERNALS
C      FDCT
C      FFTX
C      FFTY
C      A CALL TO STFLT INITIATE THE VALUES OF FILT1,FILT2,AND FILT3
C*****
      DIMENSION HR(N1,N2,N3),HI(N1,N2,N3)
*CALL DATA9
*CALL FLT
*CALL DATA7
*CALL DAT21
      LEVEL 2,HR
      CC=1.0/(IMAX*JMAX)
      IJ=N1*N2
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      DO 20 L=1,LMAX
      XR(L)=HR(I,J,L)
      20 CONTINUE
      CALL FDCT(1.0)
      DO 30 L=1,LMAX
      HI(I,J,L)=XR(L)
      30 CONTINUE
      10 CONTINUE
      DO 40 L=1,LMAX
      CALL MOVLEV(HI(1,1,L),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,1.0)
      DO 50 J=1,JMAX
      DO 50 I=1,IMAX
      FR(I,J)=FR(I,J)*FILT1(I)*FILT2(J)*FILT3(L)
      FI(I,J)=FI(I,J)*FILT1(I)*FILT2(J)*FILT3(L)
      50 CONTINUE
      CALL FFTX(-1.0)
      CALL FFTY(-1.0,CC)
      CALL MOVLEV(FR(1,1),HI(1,1,L),IJ)

```

```

40 CONTINUE
   DO 60 J=1,JMAX
   DO 60 I=1,IMAX
   DO 70 L=1,LMAX

       DO 60 I=1,IMAX
       DO 70 L=1,LMAX
       XR(L)=HI(I,J,L)
70 CONTINUE
   CALL FDCT(-1.0)
   DO 80 L=1,LMAX
   HR(I,J,L)=XR(L)
80 CONTINUE
60 CONTINUE
   RETURN
   END
*DECK CONVEC
SUBROUTINE CONVEC
C*****
C   THIS SUBROUTINE COMPUTES THE CONVECTIVE AND STRETCHING
C   TERMS AND STORES THEM IN GU,GV,GW
C   THIS ROUTINE USES AS EXTERNALS
C   COSPART
C   PARTIAL
C*****
*CALL LARGE2
*CALL LARGE3
*CALL LARGE5
*CALL BLANK
*CALL DATA9
*CALL DIM
   IJK=N1*N2*N3
C*****TERM FOR THE X-DIRECTION
   DO 10 L=1,LMAX
   DO 10 J=1,JMAX
   DO 10 I=1,IMAX
       GU(I,J,L)=U(I,J,L)*02(I,J,L)-V(I,J,L)*01(I,J,L)
       GV(I,J,L)=U(I,J,L)*03(I,J,L)-W(I,J,L)*01(I,J,L)
10 CONTINUE
   CALL PARTIAL(2,GU,N1,N2,N3)
   CALL MOVLEV(DUDX(1,1,1),GU(1,1,1),IJK)
   CALL COSPART(GV,N1,N2,N3)
   DO 20 L=1,LMAX
   DO 20 J=1,JMAX
   DO 20 I=1,IMAX
       GU(I,J,L)=GU(I,J,L)+DUDX(I,J,L)
20 CONTINUE
C*****TERM FOR THE Y-DIRECTION
   DO 30 L=1,LMAX
   DO 30 J=1,JMAX
   DO 30 I=1,IMAX
       GV(I,J,L)=V(I,J,L)*01(I,J,L)-U(I,J,L)*02(I,J,L)
       GW(I,J,L)=V(I,J,L)*03(I,J,L)-W(I,J,L)*02(I,J,L)
30 CONTINUE
   CALL PARTIAL(1,GV,N1,N2,N3)
   CALL MOVLEV(DUDX(1,1,1),GV(1,1,1),IJK)
   CALL COSPART(GW,N1,N2,N3)
   DO 40 L=1,LMAX
   DO 40 J=1,JMAX
   DO 40 I=1,IMAX
       GV(I,J,L)=GV(I,J,L)+DUDX(I,J,L)
40 CONTINUE
C*****TERM FOR THE Z-DIRECTION
   DO 50 L=1,LMAX
   DO 50 J=1,JMAX
   DO 50 I=1,IMAX
       GW(I,J,L)=W(I,J,L)*01(I,J,L)-U(I,J,L)*03(I,J,L)
       U(I,J,L)=W(I,J,L)*02(I,J,L)-V(I,J,L)*03(I,J,L)
50 CONTINUE
   CALL PARTIAL(1,GW,N1,N2,N3)

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

        CALL MOVLEV(DUDX(1,1,1),GW(1,1,1),IJK)
        CALL PARTIAL(2,U,N1,N2,N3)
        DO 60 L=1,LMAX
        DO 60 J=1,JMAX
        DO 60 I=1,IMAX
        GW(I,J,L)=GW(I,J,L)+DUDX(I,J,L)
60    CONTINUE
        RETURN
        END
*DECK COSPART
        SUBROUTINE COSPART(U,N1,N2,N3)
C*****
C      COSPART COMPUTES THE PARTIAL IN THE Z-DIRECTION OF U BY EXPANDING
C      IN FOURIER COSINE SERIES
C      THE EXTERNALS USED IN THIS SUBROUTINE ARE
C      FDCT
C      FDST
C*****
        DIMENSION U(N1,N2,N3)
*CALL BLANK
*CALL DAT21
*CALL WV
*CALL DATA9
        LEVEL 2,U
        DO 10 J=1,JMAX
        DO 10 I=1,IMAX
        DO 20 L=1,LMAX
        XR(L)=U(I,J,L)
        20    CONTINUE
        SIGN=1.0
        CALL FDCT(SIGN)
        DO 30 L=1,LMAX
        XR(L)=-XR(L)*WAVEZ(L)
        30    CONTINUE
        SIGN=-1.0
        CALL FDST(SIGN)
        DO 40 L=1,LMAX
        DUDX(I,J,L)=XR(L)
        40    CONTINUE
        10    CONTINUE
        RETURN
        END
*DECK CURLO
        SUBROUTINE CURLO(U,V,W,01,02,03,N1,N2,N3)
        DIMENSION 01(N1,N2,N3),02(N1,N2,N3),03(N1,N2,N3)
        DIMENSION U(N1,N2,N3),V(N1,N2,N3),W(N1,N2,N3)
*CALL DATA9
*CALL BLANK
        LEVEL 2,U,V,W,01,02,03
C*****
C      THIS SUBROUTINE COMPUTES THE CURL OF THE VORTICITY FIELD
C      THE EXTERNALS USED IN THIS ROUTINE ARE
C      PARTIAL
C      SINPART
C*****
C*****CURL IN THE X-DIRECTION
        IJK=N1*N2*N3
        CALL PARTIAL(2,W,N1,N2,N3)
        CALL MOVLEV(DUDX(1,1,1),01(1,1,1),IJK)
        CALL SINPART(V,N1,N2,N3)
        DO 10 L=1,LMAX
        DO 10 J=1,JMAX
        DO 10 I=1,IMAX
        01(I,J,L)=01(I,J,L)- DUDX(I,J,L)
        10    CONTINUE
C*****CURL IN THE Y-DIRECTION
        CALL SINPART(U,N1,N2,N3)
        CALL MOVLEV(DUDX(1,1,1),02(1,1,1),IJK)
        CALL PARTIAL(1,W,N1,N2,N3)
        DO 20 L=1,LMAX

```



```

DO 20 J=1,JMAX
DO 20 I=1,IMAX
02(I,J,L)=02(I,J,L)-DUDX(I,J,L)
20 CONTINUE
C*****CURL IN THE Z-DIRECTION
CALL PARTIAL(1,V,N1,N2,N3)
CALL MOVLEV(DUDX(1,1,1),03(1,1,1),IJK)
CALL PARTIAL(2,U,N1,N2,N3)
DO 30 L=1,LMAX
DO 30 J=1,JMAX
DO 30 I=1,IMAX
03(I,J,L)=03(I,J,L)-DUDX(I,J,L)
30 CONTINUE
RETURN
END
*DECK CURLU
SUBROUTINE CURLU(U,V,W,01,02,03,N1,N2,N3)
DIMENSION 01(N1,N2,N3),02(N1,N2,N3),03(N1,N2,N3)
DIMENSION U(N1,N2,N3),V(N1,N2,N3),W(N1,N2,N3)
*CALL DATA9
*CALL BLANK
LEVEL 2,U,V,W,01,02,03
C*****
C THIS ROUTINE COMPUTES THE CURL OF THE VELOCITY FIELD *
C AND STORES IT IN 01,02,03, *
C THE EXTERNALS USED IN THIS SUBROUTINE ARE *
C COSPART *
C PARTIAL *
C*****
C*****CURL IN THE X-DIRECTION
IJK=N1*N2*N3
CALL PARTIAL(2,W,N1,N2,N3)
CALL MOVLEV(DUDX(1,1,1),01(1,1,1),IJK)
CALL COSPART(V,N1,N2,N3)
DO 10 L=1,LMAX
DO 10 J=1,JMAX
DO 10 I=1,IMAX
01(I,J,L)=01(I,J,L)- DUDX(I,J,L)
10 CONTINUE
C*****CURL IN THE Y-DIRECTION
CALL COSPART(U,N1,N2,N3)
CALL MOVLEV(DUDX(1,1,1),02(1,1,1),IJK)
CALL PARTIAL(1,W,N1,N2,N3)
DO 20 L=1,LMAX
DO 20 J=1,JMAX
DO 20 I=1,IMAX
02(I,J,L)=02(I,J,L)-DUDX(I,J,L)
20 CONTINUE
C*****CURL IN THE Z-DIRECTION
CALL PARTIAL(1,V,N1,N2,N3)
CALL MOVLEV(DUDX(1,1,1),03(1,1,1),IJK)
CALL PARTIAL(2,U,N1,N2,N3)
DO 30 L=1,LMAX
DO 30 J=1,JMAX
DO 30 I=1,IMAX
03(I,J,L)=03(I,J,L)-DUDX(I,J,L)
30 CONTINUE
RETURN
END
*DECK DATARED
SUBROUTINE DATARED
C*****
C THIS SUBROUTINE COMPUTES THE STATISTICS OF THE COMPUTATION *
C USUM = PLANAR AVERAGE OF THE STREAMWISE VELOCITY *
C VSUM=PLANAR AVERAGE OF THE SPANWISE VELOCITY *
C WSUM = PLANAR AVERAGE OF THE CROSSFLOW VELOCITY *
C 01SUM = PLANAR AVERAGE OF THE STREAMWISE VORTICITY *
C 02SUM = PLANAR AVERAGE OF THE SPANWISE VORTICITY *
C 03SUM = PLANAR AVERAGE OF THE CROSSFLOW VORTICITY *
C USQ = R.M.S STREAMWISE VELOCITY *

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

C      VSQ = R.M.S. OF THE SPANWISE VELOCITY ETC...
C      UVSTRES = PLANAR AVERAGE OF U3W3
C      PLOVALE = VOLUME AVERAGE OF THE TOTAL ENERGY
C      ENERGY = INTEGRAL OF THE TURBULENCE ENERGY
C*****
*CALL DEL
*CALL MEANVOR
*CALL PR
*CALL LARGE2
*CALL LARGE3
*CALL LARGE5
*CALL DAT21
*CALL DATA9
*CALL BLANK
*CALL DIM
      DIMENSION USM(33),VSM(33),US(33),VS(33),WS(33),O1S(33),O2S(33)
1      ,O3S(33),ES(33),ENS(33),ZO(33)
      DIMENSION DUMSP(33)
      COMMON/DATCNT/ IDATCNT
      IDATCNT=IDATCNT+1
      LMAXM1=LMAX-1
      C3=1./LMAXM1
      CNORM3=1./(IMAX*JMAX)
      PRINT 1100
      UTOT=0.
      VTOT=0.
      WTOT=0.
      O1TOT=0.
      O2TOT=0.
      O3TOT=0.
      OVRALE=0.
      TOTENER=0.
      TOTENST=0.
      DO 100 L=1,LMAX
      USUM=0.
      VSUM=0.
      WSUM=0.
      O1SUM=0.
      O2SUM=0.
      O3SUM=0.
      USQ=0.
      VSQ=0.
      WSQ=0.
      O1SQ=0.
      O2SQ=0.
      O3SQ=0.
      ENERGY=0.
      ENSTROP=0.
      UVSTRES=0.
      PLOVALE=0.
      DO 110 J=1,JMAX
      DO 110 I=1,IMAX
      USUM=USUM+U(I,J,L)
      VSUM=VSUM+V(I,J,L)
      WSUM=WSUM+W(I,J,L)
      O1SUM=O1SUM+O1(I,J,L)
      O2SUM=O2SUM+O2(I,J,L)
      O3SUM=O3SUM+O3(I,J,L)
110  CONTINUE
      USUM=USUM*CNORM3
      VSUM=VSUM*CNORM3
      WSUM=WSUM*CNORM3
      O1SUM=O1SUM*CNORM3
      O2SUM=O2SUM*CNORM3
      O3SUM=O3SUM*CNORM3
      GW(1,1,L)=USUM
      GW(2,1,L)=VSUM
      GW(3,1,L)=WSUM
      DO 160 J=1,JMAX
      DO 160 I=1,IMAX

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

USQ=USQ+(U(I,J,L)-USUM)**2
VSQ=VSQ+(V(I,J,L)-VSUM)**2
WSQ=WSQ+(W(I,J,L)-WSUM)**2
O1SQ=O1SQ+(O1(I,J,L)-O1SUM)**2
O2SQ=O2SQ+(O2(I,J,L)-O2SUM)**2
O3SQ=O3SQ+(O3(I,J,L)-O3SUM)**2
UVSTRES=UVSTRES+(U(I,J,L)-USUM)*(W(I,J,L)-WSUM)
PLOVALE=PLOVALE+(U(I,J,L)**2+V(I,J,L)**2+W(I,J,L)**2)
160 CONTINUE
USQ=USQ*CNORM3
VSQ=VSQ*CNORM3
WSQ=WSQ*CNORM3
O1SQ=O1SQ*CNORM3
O2SQ=O2SQ*CNORM3
O3SQ=O3SQ*CNORM3
ENERGY=(USQ+VSQ+WSQ)*0.5
ENSTROP=(O1SQ+O2SQ+O3SQ)*0.5
UVSTRES=UVSTRES*CNORM3
USQ=SQRT(USQ)
VSQ=SQRT(VSQ)
WSQ=SQRT(WSQ)
O1SQ=SQRT(O1SQ)
O2SQ=SQRT(O2SQ)
O3SQ=SQRT(O3SQ)
US(L)=USQ
VS(L)=VSQ
WS(L)=WSQ
O1S(L)=O1SQ
O2S(L)=O2SQ
O3S(L)=O3SQ
USM(L)=USUM
VSM(L)=VSUM
ES(L)=ENERGY
ENS(L)=ENSTROP
GW(4,1,L)=O1SUM
GW(5,1,L)=O2SUM
GW(6,1,L)=O3SUM
GV(1,1,L)=USQ
GV(2,1,L)=WSQ
XI(L)=UVSTRES
CC=1.
IF(L.EQ. 1) CC=0.5
IF(L.EQ. LMAX) CC=0.5
OVRALE=OVRALE+PLOVALE*CC*0.5
UTOT=UTOT+USUM*CC
VTOT=VTOT+VSUM*CC
WTOT=WTOT+WSUM*CC
O1TOT=O1TOT+O1SUM*CC
O2TOT=O2TOT+O2SUM*CC
O3TOT=O3TOT+O3SUM*CC
TOTENER=TOTENER+ENERGY*CC
TOTENST=TOTENST+ENSTROP*CC
100 CONTINUE
UTOT=UTOT*C3
VTOT=VTOT*C3
WTOT=WTOT*C3
O1TOT=O1TOT*C3
O2TOT=O2TOT*C3
O3TOT=O3TOT*C3
TOTENER=TOTENER*C3
TOTENST=TOTENST*C3
DELU=GW(1,1,LMAX)-GW(1,1,1)
DELU=1./DELU
THETA=(0.25-(GW(1,1,1)*DELU)**2)*0.5
DO 170 L=2,LMAXM1
THETA=THETA+(0.25-(GW(1,1,L)*DELU)**2)
170 CONTINUE
THETA=THETA+(0.25-(GW(1,1,LMAX)*DELU)**2)*0.5
THETA=THETA*DELTAZ
DO 300 L=1,LMAX

```

```

Z0(L)=(L-((LMAX-1)/2+1))*DELTAZ/THETA
USM(L)=USM(L)*DELU*2.0
VSM(L)=VSM(L)*DELU
GW(5,1,L)=GW(5,1,L)*DELU*THETA
XI(L)=XI(L)*(DELU**2)
US(L)=US(L)*DELU
VS(L)=VS(L)*DELU
WS(L)=WS(L)*DELU
O3S(L)=O3S(L)*(DELU*THETA)
O1S(L)=O1S(L)*(DELU*THETA)
O2S(L)=O2S(L)*(DELU*THETA)
ES(L)=ES(L)*DELU**2
ENS(L)=ENS(L)*(DELU*THETA)**2
PRINT 3000,USM(L),VSM(L),XI(L),GW(5,1,L),US(L),VS(L),WS(L),
1 O1S(L),O2S(L),O3S(L),ES(L),ENS(L),Z0(L)
300 CONTINUE
WRITE(8) USM,VSM,XI,US,VS,WS,O1S,O2S,O3S,ES,ENS,Z0,THETA
PRINT 1700,THETA
PRINT 1200
PRINT 1000,UTOT,VTOT,WTOT,O1TOT,O2TOT,O3TOT,TOTENER,TOTENST
PRINT 2400,OVRALE
2400 FORMAT(1X,* OVER ALL ENERGY IN COMPUTATION BOX =*,1PE15.7)
DO 180 L=1,LMAX
DO 180 I=1,IMAX
GU(I,1,L)=0.
GU(I,2,L)=0.
GU(I,3,L)=0.
GU(I,4,L)=0.
180 CONTINUE
CLOY=1./FLOAT(JMAX)
DO 190 L=1,LMAX
DO 190 J=1,JMAX
DO 190 I=1,IMAX
GU(I,1,L)=GU(I,1,L)+O2(I,J,L)*CLOY
GU(I,2,L)=GU(I,2,L)+U(I,J,L)*CLOY
GU(I,3,L)=GU(I,3,L)+W(I,J,L)*CLOY
190 CONTINUE
DO 230 L=1,LMAX
DO 230 I=1,IMAX
GU(I,2,L)=GU(I,2,L)-GW(1,1,L)
GU(I,3,L)=GU(I,3,L)-GW(3,1,L)
230 CONTINUE
PRINT 2200
2200 FORMAT(1H1,1X,* LINE AVERAGE OF VORTICITY*)
PRINT 2300,(((GU(I,1,L),I= 1,16 ),L),L=1,LMAX)
IF(CCPD .NE.1.) GO TO 240
PRINT 2500
2500 FORMAT(1H1,1X,* LINE AVERAGE OF U-COMPONENT *)
PRINT 2300,(((GU(I,2,L),I= 1,16 ),L),L=1,LMAX)
PRINT 2500
PRINT 2300,(((GU(I,2,L),I=17,IMAX),L),L=1,LMAX)
PRINT 2600
2600 FORMAT(1H1,1X,* LINE AVERAGE OF W-COMPONENT *)
PRINT 2300,(((GU(I,3,L),I= 1,16 ),L),L=1,LMAX)
PRINT 2600
PRINT 2300,(((GU(I,3,L),I=17,IMAX),L),L=1,LMAX)
2300 FORMAT(1X,16F8.3,I3)
240 CONTINUE
PRINT 2000
DO 250 L=14,20
DO 260 I=1,IMAX
XR(I)=GU(I,1,L)
XI(I)=0.
260 CONTINUE
CALL FFT(XR,XI,IMAX,-1)
IF(IDATCNT.EQ.1) DUMSP(L)=SQRT(XR(2)**2+XI(2)**2)
IF(DUMSP(L).LT.0.0000001) GO TO 250
DO 270 I=1,IMAX
XR(I)=SQRT(XR(I)**2+XI(I)**2)/DUMSP(L)
270 CONTINUE

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS P⁰⁰

```

      PRINT 1800,L,(XR(I),I=1,8)
250  CONTINUE
1800  FORMAT(1X,*WV02*,I5,1P8E14.6)
1000  FORMAT(1P8E15.7)
1100  FORMAT(2X,*USUM*,6X,*VSUM*,5X,*UWSTR*,5X,*02SUM*,7X,*USQ*,7X,*VSQ*
      1,7X,*WSQ*,6X,*01SQ*,6X,*02SQ*,6X,*03SQ*,5X,*ENERGY*,4X,*ENSTROP*,3
      2X,*PLANE*)
1200  FORMAT(///,1X,* UTOT IN X-Y   VTOT IN X-Y   WTOT IN X-Y   01TOT
      1IN X-Y   02TOT IN X-Y   03TOT IN X-Y *)
1300  FORMAT(1X,* USUN IN Y-Z   VSUM IN Y-Z   WSUM IN Y-Z   01SUM IN Y
      1-Z   02SUM IN Y-Z   03SUM IN Y-Z *)
1400  FORMAT(///,1X,* UTOT IN Y-Z   VTOT IN Y-Z   WTOT IN Y-Z   01TOT
      1IN Y-Z   02TOT IN Y-Z   03TOT IN Y-Z *)
1500  FORMAT(1X,* USUN IN Z-X   VSUM IN Z-X   WSUM IN Z-X   01SUM IN Z
      1-X   02SUM IN Z-X   03SUM IN Z-X *)
1600  FORMAT(///,1X,* UTOT IN Z-X   VTOT IN Z-X   WTOT IN Z-X   01TOT
      1IN Z-X   02TOT IN Z-X   03TOT IN Z-X *)
1700  FORMAT(1X,* MOMENTUM THICKNESS *,1PE15.7)
2000  FORMAT(1H1)
3000  FORMAT(1P13E10.2)
C     TEST THE DIVERGENCE OF THE VELOCITY AND VORTICITY FIELDS
C     CALL DIV
C     TEST THE SOLUTION OF THE POISSON EQUATIONS ,I.E. THAT
C     THE CURL OF OUR VELOCITY FIELD IS EQUAL TO THE VORTICITY FIELD
C     CALL CURLU(U,V,W,GU,GV,GW,N1,N2,N3)
      ERRMAX1=0.
      ERRMAX2=0.
      ERRMAX3=0.
      DO 17 L=1,LMAX
      DO 17 J=1,JMAX
      DO 17 I=1,IMAX
      GU(I,J,L)=ABS(GU(I,J,L)-01(I,J,L))
      GV(I,J,L)=ABS(GV(I,J,L)-02(I,J,L))
      GW(I,J,L)=ABS(GW(I,J,L)-03(I,J,L))
      IF (GU(I,J,L) .GT. ERRMAX1) ERRMAX1=GU(I,J,L)
      IF (GV(I,J,L) .GT. ERRMAX2) ERRMAX2=GV(I,J,L)
      IF (GW(I,J,L) .GT. ERRMAX3) ERRMAX3=GW(I,J,L)
17  CONTINUE
      PRINT 1110,ERRMAX1,ERRMAX2,ERRMAX3
1110  FORMAT(1X,* ERRMAX1 =*,E15.7,*ERRMAX2 =*,E15.7,* ERRMAX3 =*,E15.7)
      RETURN
      END
*DECK DIV
      SUBROUTINE DIV
C*****
C     THIS ROUTINE TESTS THE DIVERGENCE OF THE VELOCITY FIELD
C     AND THE DIVERGENCE OF THE VORTICITY FIELD .
C*****
*CALL LARGE2
*CALL LARGE3
*CALL LARGE5
*CALL BLANK
*CALL DATA9
*CALL DIM
      IJK=N1*N2*N3
      CALL PARTIAL(1,U,N1,N2,N3)
      CALL MOVLEV(DUDX(1,1,1),GU(1,1,1),IJK)
      CALL PARTIAL(2,V,N1,N2,N3)
      CALL MOVLEV(DUDX(1,1,1),GV(1,1,1),IJK)
      CALL SINPART(W,N1,N2,N3)
      DIVMAX=0.
      DO 1 L=1,LMAX
      DO 1 J=1,JMAX
      DO 1 I=1,IMAX
      DUM=ABS(GU(I,J,L)+GV(I,J,L)+DUDX(I,J,L))
      IF (DUM .GT. DIVMAX) DIVMAX=DUM
1  CONTINUE
      PRINT 1100,DIVMAX
      CALL PARTIAL(1,01,N1,N2,N3)
      CALL MOVLEV(DUDX(1,1,1),GU(1,1,1),IJK)

```

```

CALL PARTIAL(2,02,N1,N2,N3)
CALL MOVLEV(DUDX(1,1,1),GV(1,1,1),IJK)
CALL COSPART(03,N1,N2,N3)
DIVMAX=0.
DO 2 L=1,LMAX
DO 2 J=1,JMAX
DO 2 I=1,IMAX
DUM=ABS(GU(I,J,L)+GV(I,J,L)+DUDX(I,J,L))
IF (DUM.GT. DIVMAX) DIVMAX=DUM
2 CONTINUE
PRINT 1200,DIVMAX
1100 FORMAT(1X,* MAXIMUM VELOCITY DIVERGENCE =*,E15.7)
1200 FORMAT(1X,* MAXIMUM VORTICITY DIVERGENCE =*,E15.7)
RETURN
END
*DECK EDVIS
SUBROUTINE EDVIS(COEF2,E,N1,N2,N3)
DIMENSION E(N1,N2,N3)
C*****
C THIS SUBROUTINE COMPUTES THE EDDY VISCOSITY AND STORES IT IN E *
C*****
*CALL LARGE5
*CALL DATA9
DO 3 L=1,LMAX
DO 3 J=1,JMAX
DO 3 I=1,IMAX
E(I,J,L)=01(I,J,L)**2+02(I,J,L)**2+03(I,J,L)**2
E(I,J,L)=SQRT(E(I,J,L))*COEF2
3 CONTINUE
RETURN
END
*DECK FDCT
SUBROUTINE FDCT(SIGN)
C*****
C FDCT COMPUTES THE FAST DISCRETE COSINE TRANSFORM OF THE VARIABLE *
C XR AND STORES IN XR *
C*****
*CALL DAT21
*CALL DATA9
LM1=LMAX-1
CC=2./FLOAT(LM1)
LL=2*LM1
XR(1)=XR(1)/2.
XR(LMAX)=XR(LMAX)/2.
LP1=LMAX+1
DO 1 L=LP1,LL
XR(L)=0.
1 CONTINUE
DO 3 L=1,LL
XI(L)=0.
3 CONTINUE
ISN=-SIGN
CALL FFT(XR,XI,LL,ISN)
IF (SIGN.GT. 0.) GO TO 200
DO 100 L=1,LMAX
XR(L)=XR(L)*CC
100 CONTINUE
200 CONTINUE
RETURN
END
*DECK FDST
SUBROUTINE FDST(SIGN)
C*****
C FDST COMPUTES THE FAST DISCRETE SINE TRANSFORM OF THE VARIABLE *
C XR AND STORES IT IN XR *
C*****
*CALL DAT21
*CALL DATA9
LM1=LMAX-1
CC=2./FLOAT(LM1)

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

LL=2*LM1
XR(1)=0.
XR(LMAX)=0.
LP1=LMAX+1
DO 1 L=LP1,LL
  XR(L)=0.
1 CONTINUE
DO 3 L=1,LL
  XI(L)=0.
3 CONTINUE
ISN=-SIGN
CALL FFT(XR,XI,LL,ISN)
IF (SIGN.GT. 0.) GO TO 200
DO 100 L=1,LMAX
  XI(L)=XI(L)*CC
100 CONTINUE
200 CONTINUE
DO 2 L=1,LMAX
  XR(L)=-SIGN*XI(L)
2 CONTINUE
RETURN
END
*DECK FFT

```

	IDENT FFT (A,B,N,ISN)	FFT2C 2
	ENTRY FFT	FFT2C 3
*	RADIX 2 COMPLEX FAST FOURIER TRANSFORM, COMPUTED IN PLACE.	FFT2C 4
*	SEE ON COMPUTING THE FAST FOURIER TRANSFORM, R. SINGLETON,	FFT2C 5
*	COMM. ACM, V.10, N.10, PP.647-654, OCT. 1967.	FFT2C 6
*	ARRAY A CONTAINS THE REAL COMPONENT OF THE DATA AND RESULT,	FFT2C 7
*	ARRAY B CONTAINS THE IMAGINARY COMPONENT.	FFT2C 8
*	N, THE NUMBER OF COMPLEX DATA VALUES,	FFT2C 9
*	MUST BE A POWER OF 2 AND GREATER THAN 1	FFT2C 10
*	THE SIGN OF ISN IS THE SIGN OF THE EXPONENTIAL IN THE TRANSFORM.	FFT2C 11
*	THE MAGNITUDE OF ISN IS THE INCREMENT SIZE FOR INDEXING	FFT2C 12
*	A AND B, AND IS ONE IN THE USUAL CASE.	FFT2C 13
*	DATA MAY ALTERNATIVELY BE STORED FORTRAN COMPLEX	FFT2C 14
*	IN A SINGLE ARRAY, IN WHICH CASE THE MAGNITUDE	FFT2C 15
*	OF ISN IS TWO AND ADDRESS B IS A(2), I.E.	FFT2C 16
*	CALL FFT2(A,A(2),N,2)	FFT2C 17
*	INSTEAD OF	FFT2C 18
*	CALL FFT2(A,B,N,1)	FFT2C 19
*	PROGRAM CONTAINS SINE TABLE FOR MAXIMUM N OF 32768	FFT2C 20
*	6400 TIME FOR N=1024, 220 M.SEC.	FFT2C 21
*	6400 TIME FOR N=2**M IS 21.5*N*M MICRO-SEC.	FFT2C 22
*	6600 TIME FOR N=1024, 44 M.SEC.	FFT2C 23
*	6600 TIME FOR N=2**M IS 4.3*N*M MICRO-SEC.	FFT2C 24
*	RMS ERROR FOR TRANSFORM-INVERSE IS LESS THAN 1.3E-13	FFT2C 25
*	FOR N=32768 OR SMALLER.	FFT2C 26
*	FORTRAN 2.3 SUBROUTINE	FFT2C 27
*	BY R. C. SINGLETON, STANFORD RESEARCH INSTITUTE, NOV. 1968	FFT2C 28
L100	SX0 B3 NN	FFT2C 29
	SB4 B0 KK=0	FFT2C 30
	SB3 B3-B7 NN=NN-INC	FFT2C 31
	AX0 1 KSPAN=NN/2	FFT2C 32
	SB5 B0 K2=0	FFT2C 33
	SB6 X0	FFT2C 34
	SX1 B5 K2=K2	FFT2C 35
	EQ B6,B7,FFT IF(KSPAN.EQ. INC) RETURN	FFT2C 36
L110	SB4 B3-B4 KK=NN-KK	FFT2C 37
	SB5 B3-B5 K2=NN-K2	FFT2C 38
	SA2 B1+B4 EXCHANGE A(KK),A(K2) AND B(KK),B(K2)	FFT2C 39
	SA3 B1+B5	FFT2C 40
	SA4 B2+B4	FFT2C 41
	NX7 X2	FFT2C 42
	SA5 B2+B5	FFT2C 43
	NX6 X3	FFT2C 44
	SA7 A3	FFT2C 45
	SA6 A2	FFT2C 46
	NX7 X4	FFT2C 47
	NX6 X5	FFT2C 48

	SA7	A5		FFT2C	49
	SA6	A4		FFT2C	50
L120	LT	B6,B4,L110	END OF EXCHANGE	FFT2C	51
	SB4	B4+B7	IF(KSPAN .LT. KK) GO TO L110	FFT2C	52
	SB5	B6+B5	KK=KK+INC	FFT2C	53
	SA2	B1+B4	K2=KSPAN+K2	FFT2C	54
	SA3	B1+B5	EXCHANGE A(KK),A(K2) AND B(KK),B(K2)	FFT2C	55
	SA4	B2+B4		FFT2C	56
	NX7	X2		FFT2C	57
	SA5	B2+B5		FFT2C	58
	NX6	X3		FFT2C	59
	SA7	A3		FFT2C	60
	SA6	A2		FFT2C	61
	NX7	X4	K=KSPAN	FFT2C	62
	SX0	B6		FFT2C	63
	NX6	X5		FFT2C	64
	SA7	A5		FFT2C	65
L130	SA6	A4	END OF EXCHANGE	FFT2C	66
	AX0	1	K=K/2	FFT2C	67
	IX1	X1-X0	K2=K2-K	FFT2C	68
	PL	X1,L130	IF(K2 .GE. 0) GO TO L130	FFT2C	69
	LX0	1	K=K+K	FFT2C	70
	SB4	B4+B7	KK=KK+INC	FFT2C	71
	IX1	X1+X0	K2=K2+K	FFT2C	72
	SB5	X1	K2=K2	FFT2C	73
	GE	B5,B4,L110	IF(K2 .GE. KK) GO TO L110	FFT2C	74
FFT	LT	B4,B6,L120	IF(KK .LT. KSPAN) GO TO L120	FFT2C	75
	SB1	X1		FFT2C	76
	SA1	A1+1	INSR1	INSR1	1
	SB2	X1	INSR1	INSR1	2
	SA1	A1+1	INSR1	INSR1	3
	SB3	X1	INSR1	INSR1	4
	SA1	A1+1	INSR1	INSR1	5
	SB4	X1	INSR1	INSR1	6
	SA4	B4	INSR1	INSR1	7
	MX2	1	FFT2C	FFT2C	77
	SA5	L60	FFT2C	FFT2C	78
	SA3	B3	FFT2C	FFT2C	79
	LX2	57	FFT2C	FFT2C	80
	PX7	X3	FFT2C	FFT2C	81
	BX6	-X2*X5	FFT2C	FFT2C	82
	PL	X4,L110	FFT2C	FFT2C	83
	BX6	X2+X5	FFT2C	FFT2C	84
	BX4	-X4	FFT2C	FFT2C	85
L10	LX3	32	FFT2C	FFT2C	86
	SA6	A5	FFT2C	FFT2C	87
	NX0	B5,X3	FFT2C	FFT2C	88
	PX2	X4	FFT2C	FFT2C	89
	SB7	X4	FFT2C	FFT2C	90
	DX7	X2*X7	FFT2C	FFT2C	91
	SA1	B5+S	FFT2C	FFT2C	92
	SB3	X7	FFT2C	FFT2C	93
	SB6	X7	FFT2C	FFT2C	94
	EQ	L40	FFT2C	FFT2C	95
L20	SA3	CD	FFT2C	FFT2C	96
	RX4	X2*X1	FFT2C	FFT2C	97
	RX7	X2*X0	FFT2C	FFT2C	98
	RX5	X3*X0	FFT2C	FFT2C	99
	RX6	X3*X1	FFT2C	FFT2C	100
	RX4	X4-X5	FFT2C	FFT2C	101
	RX6	X6+X7	FFT2C	FFT2C	102
	NX5	X4	FFT2C	FFT2C	103
	RX7	X1-X6	FFT2C	FFT2C	104
	RX0	X0+X5	FFT2C	FFT2C	105
	NX1	X7	FFT2C	FFT2C	106
L30	SB5	B6+B4	FFT2C	FFT2C	107
	SA2	B1+B4	FFT2C	FFT2C	108
	SA3	B1+B5	FFT2C	FFT2C	109
	SA4	B2+B4	FFT2C	FFT2C	110
			FFT2C	FFT2C	111

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

	RX6	X2+X3			FFT2C112
	SA5	B2+B5		B(K2)	FFT2C113
	RX2	X2-X3		RE	FFT2C114
	SA6	A2		A(KK)	FFT2C115
	RX7	X4+X5			FFT2C116
	RX3	X1*X2		CN*RE	FFT2C117
	RX4	X4-X5		IM	FFT2C118
	SA7	A4		B(KK)	FFT2C119
	RX5	X0*X4		SN*IM	FFT2C120
	RX2	X0*X2		SN*RE	FFT2C121
	RX6	X3-X5			FFT2C122
	RX4	X1*X4		CN*IM	FFT2C123
	SA6	A3		A(K2)	FFT2C124
	RX7	X2+X4			FFT2C125
	SB4	B6+B5		KK=KSPAN+K2	FFT2C126
	SA7	A5		B(K2)	FFT2C127
	LT	B4,B3,L30		IF(KK .LT. NN) GO TO L30	FFT2C128
	SB5	B4-B3		K2=KK-NN	FFT2C129
	BX1	-X1		CN=-CN	FFT2C130
	SB4	B6-B5		KK=KSPAN-K2	FFT2C131
	LT	B5,B4,L30		IF(K2 .LT. KK) GO TO L30	FFT2C132
	SB4	B4+B7		KK=KK+INC	FFT2C133
	SA2	SD			FFT2C134
	LT	B4,B5,L20		IF(KK .LT. K2) GO TO L20	FFT2C135
L40	SB4	B0		KK=0	FFT2C136
	SX5	B6			FFT2C137
	AX5	1		KSPAN=KSPAN/2	FFT2C138
	SB6	X5			FFT2C139
L50	SB5	B6+B4		K2=KSPAN+KK	FFT2C140
	SA2	B1+B4		A(KK)	FFT2C141
	SA3	B1+B5		A(K2)	FFT2C142
	SA4	B2+B4		B(KK)	FFT2C143
	RX6	X2+X3			FFT2C144
	SA5	B2+B5		B(K2)	FFT2C145
	RX7	X2-X3			FFT2C146
	SA6	A2		A(KK)	FFT2C147
	SA7	A3		A(K2)	FFT2C148
	RX6	X4+X5			FFT2C149
	SB4	B6+B5		KK=KSPAN+K2	FFT2C150
	RX7	X4-X5			FFT2C151
	SA6	A4		B(KK)	FFT2C152
	SA7	A5		B(K2)	FFT2C153
	LT	B4,B3,L50		IF(KK .LT. NN) GO TO L50	FFT2C154
	EQ	B6,B7,L100		IF(KSPAN .EQ. INC) GO TO L100	FFT2C155
	SA1	A1		S(M)	FFT2C156
	SB4	B7		KK=INC	FFT2C157
	RX6	X1*X1			FFT2C158
	SA1	A1+1		M=M+1, S(M)	FFT2C159
	FX6	X6+X6			FFT2C160
	SA3	ONE			FFT2C161
	SA6	CD		CD=2*S(M)**2	FFT2C162
L60	BX0	X1		SN=SD	FFT2C163
	RX6	X3-X6		CN=1.0-CD	FFT2C164
	BX7	X0			FFT2C165
	NX1	X6			FFT2C166
	SA7	SD			FFT2C167
	EQ	L30		GO TO L30	FFT2C168
S	DATA	9.5873799095977346E-5			FFT2C169
	DATA	1.9174759731070331E-4			FFT2C170
	DATA	3.8349518757139559E-4			FFT2C171
	DATA	7.6699031874270453E-4			FFT2C172
	DATA	1.5339801862847656E-3			FFT2C173
	DATA	3.0679567629659763E-3			FFT2C174
	DATA	6.1358846491544754E-3			FFT2C175
	DATA	1.2271538285719926E-2			FFT2C176
	DATA	2.4541228522912288E-2			FFT2C177
	DATA	4.9067674327418014E-2			FFT2C178
	DATA	9.8017140329560602E-2			FFT2C179
	DATA	1.9509032201612827E-1			FFT2C180
	DATA	3.8268343236508977E-1			FFT2C181

```

ONE      DATA    0.7071067811865475      FFT2C182
CD       DATA    1.0                      FFT2C183
SD                               FFT2C184
                               FFT2C185

      END
*DECK FFTX
      SUBROUTINE FFTX(SIGN)
C*****
C FAST FOURIER TRANSFORM IN X-DIRECTION
C*****
*CALL DATA9
*CALL DATA7
*CALL DAT21
      ISN=-SIGN
      IF (SIGN .LT. 0.) GO TO 3
      DO 1 J=1,JMAX
      DO 1 I=1,IMAX
      FI(I,J)=0.
      1 CONTINUE
      3 CONTINUE
      DO 100 J=1,JMAX
      DO 110 I=1,IMAX
      XR(I)=FR(I,J)
      XI(I)=FI(I,J)
      110 CONTINUE
      CALL FFT(XR,XI,IMAX,ISN)
      DO 120 I=1,IMAX
      FR(I,J)=XR(I)
      FI(I,J)=XI(I)
      120 CONTINUE
      100 CONTINUE
      RETURN
      END
*DECK FFTY
      SUBROUTINE FFTY(SIGN,COEF3)
C*****
C FAST FOURIER TRANSFORM IN Y-DIRECTION
C*****
*CALL DATA9
*CALL DATA7
*CALL DAT21
      ISN=-SIGN
      DO 100 I=1,IMAX
      DO 110 J=1,JMAX
      XR(J)=FR(I,J)
      XI(J)=FI(I,J)
      110 CONTINUE
      CALL FFT(XR,XI,JMAX,ISN)
      IF (SIGN .LT. 0.) GO TO 200
      DO 120 J=1,JMAX
      FR(I,J)=XR(J)
      FI(I,J)=XI(J)
      120 CONTINUE
      GO TO 100
      200 DO 130 J=1,JMAX
      FR(I,J)=XR(J)*COEF3
      FI(I,J)=XI(J)*COEF3
      130 CONTINUE
      100 CONTINUE
      RETURN
      END
*DECK FIX
      SUBROUTINE FIX(IM1,I,IP1,IMAX)
      IM1=I-1
      IP1=I+1
      IF(I .EQ. 1) IM1=IMAX
      IF(I .EQ. IMAX) IP1=1
      RETURN
      END
*DECK INICON

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

SUBROUTINE INICON(C,COF,DT,UR,VR,WR,UI,VI,WI,L1,L2,L3)
REAL NDIV,N12,NSQR
DIMENSION UR(L1,L2,L3),VR(L1,L2,L3),WR(L1,L2,L3)
DIMENSION UI(L1,L2,L3),VI(L1,L2,L3),WI(L1,L2,L3)
C*****
C THIS SUBROUTINE INITIATES THE PROGRAM. FOR STARTING PROBLEM, THE INI-
C TIAL FIELD IS GENERATED. FOR CONTINUATION PROBLEM, THE DATA STORED
C ON TAPE AT THE END OF THE PREVIOUS RUN ARE READ IN.
C*****
C-----NSTART=1 STARTING FROM TIME STEP=0
C-----NSTART=2 CONTINUED FROM PREVIOUS RUN
C-----IMAX=MAXIMUM MESH NUMBER IN X-DIRECTION
C-----JMAX=MAXIMUM MESH NUMBER IN Y-DIRECTION
C-----LMAX=MAXIMUM MESH NUMBER IN Z-DIRECTION
C-----TSTART=STARTING TIME STEP
C-----TEND=ENDING TIME STEP
C-----DELTA= MESH SIZE
C-----DT=TIME STEP
C-----C=MODEL CONSTANT
C-----NAVG=DELTA(AVERAGING)/DELTA(MESH)
C-----ANISO= R IN EQUATION (5.16B)
C-----GAMMA=STRAIN RATE
REAL NAVG
COMMON/LARGE1/EN(1024),EN1(1024),WN(2048)
LEVEL 2,UR,VR,WR,UI,VI,WI
LEVEL 2,EN,EN1,WN
*CALL DATA7
READ 4, DELTA,DT,C, NAVG,ANISO,UTM,GAMMA
IMAX=16
JMAX=16
LMAX=16
IJK=IMAX*JMAX*LMAX
IJ=IMAX*JMAX
IMM1=IMAX-1
IMP1=IMAX+1
NMESH=IMAX
TDIV=1.0/(IMAX*JMAX*LMAX)
NHALF=NMESH/2
NHPI1=NHALF+1
HALF=FLOAT(NHALF)
NM1=NMESH-1
RISO=3./(3.+ANISO)
TEMP=ANISO/3.
RANISO=SQRT(TEMP)
CC=1.
TFAC=IMAX*JMAX*LMAX
FAC=SQRT(TFAC)
COEF10=3.1415926535898/NHALF
COEF11=COEF10
COEF12=3.1415926535898*2.
CONST=COEF10/ DELTA
COEF15=COEF12*FAC
PI1=COEF10
PI2=PI1*2.
COF=1.0
NCONT=1
DO 2 M=1,25
Y9=RGEX(X9)
2 CONTINUE
C-----ENERGY SPECTRUM DATA
C----- 0.1 INTERVAL UP TO 1.0 THEN .5 INTERVALL UP TO 6.0
C-----EN IS THE ENERGY SPECTRUM FOR THE ISOTROPIC PART. EN1 IS FOR THE
C ANISOTROPIC PART.
PRINT 1960
WN(1)=0.1
DO 1900 M=2,10
1900 WN(M)=0.1+WN(M-1)
DO 1950 M=11,24
1950 WN(M)=0.5+WN(M-1)
1960 FORMAT(/5X,*WAVE NUMBER*,/)

```

```

      PRINT 4, (WN(M),M=1,24)
      PRINT 2000
2000  FORMAT(/5X,*UNFILTERED SPECTRUM*,/)
      DO 3 M=1,24,8
      M7=M+7
      READ 4, (EN(MM),MM=M,M7)
      PRINT 702,(EN(MM),MM=M,M7)
      3  CONTINUE
      DO 503 M=1,24,8
      M7=M+7
      READ 4, (EN1(MM),MM=M,M7)
      PRINT 702,(EN1(MM),MM=M,M7)
503   CONTINUE
      4  FORMAT (8E10.4)
      DELAVG=(DELTA*NAVG)**2/12.0
      PAI=3.1415926535898
      DO 2100 M=1,24
      EXPF=EXP(-DELAVG*WN(M)*WN(M))
      EN(M)=EN(M)*EXPF
2100  EN1(M)=EN1(M)*EXPF
      PRINT 2200
2200  FORMAT(/5X,*FILTERED SPECTRUM*,/)
      PRINT 702,(EN(M),M=1,24)
      PRINT 702,(EN1(M),M=1,24)
      DO 5 L=1,LMAX
      DO 5 J=1,JMAX
      DO 5 I=1,IMAX
      UR(I,J,L)=0.
      VR(I,J,L)=0.
      WR(I,J,L)=0.
      UI(I,J,L)=0.
      VI(I,J,L)=0.
      WI(I,J,L)=0.
      5  CONTINUE
      DO 40 L=1,NHALF
      LL=L
      N3=L-1
      N3S=N3**2
      DO 30 J=1,NM1
      JINDEX=J/NHALF
      JJ=J+NHP1-JINDEX*JMAX
      N2=J-NHALF
      N2S=N2**2
      DO 20 I=1,NM1
      IINDEX=I/NHALF
      II=I+NHP1-IINDEX*IMAX
      N1=I-NHALF
      N1S=N1**2
      NSQR=N1S+N2S+N3S
      IF (NSQR .LT. 0.1) GO TO 20
      WAVN=SQRT(NSQR)
      NDIV=1./WAVN
      N12=N1S+N2S
      IF (N12 .LT. 0.1) NCONT=2
      IF (IABS(N1) .EQ. NHALF .AND. IABS(N2) .EQ. NHALF) NCONT=2
C-----GET FOURIER AMPLITUDE OF THE INITIAL FIELD AD DESCRIBED IN SEC 4.4
      X=CONST*WAVN
      NREG=X+1.
      GO TO (310,315,315,315,315,315,315) NREG
310  M=X/0.1
      YM=X-0.1*M
      M1=M+1
      ED=EN(M1)-EN(M)
      ENERGY=EN(M)+ED*YM*10.
      EDA=EN1(M1)-EN1(M)
      EANISO=EN1(M)+EDA*YM*10.
      GO TO 320
315  M=(X-1.)*2.
      YM=X-1.-0.5*M
      M=M+10

```

```

M1=M+1
ED=EN(M1)-EN(M)
ENERGY=EN(M)+ED*YM*2.
EDA=EN1(M1)-EN1(M)
EANISO=EN1(M)+EDA*YM*2.
320 QS=ENERGY*RISO/(COEF15*X**2)
QN=SQRT(QS)
QSA=EANISO*RISO/(COEF15*X**2)
QNA=SQRT(QSA)
C-----CHANGE WAVE NUMBER VECTOR TO SATISFY NUMERICSL DIV FREE
C-----R1,R2 AND R3 ARE THE MODIFIED WAVE NUMBER
IF(NCONT.EQ.2)GO TO 340
ARG1=PI1*N1
ARG2=PI2*N1
R1=ARG1/DELTA
ARG1=PI1*N2
ARG2=PI2*N2
R2=ARG1/DELTA
ARG1=PI1*N3
ARG2=PI2*N3
R3=ARG1/DELTA
R1S=R1**2
R2S=R2**2
R3S=R3**2
R12S=R1S+R2S
RSQ=R12S+R3S
IF(NCONT.EQ.2) GO TO 340
R12=SQRT(R12S)
R12DIV=1./R12

R12=SQRT(R12S)
R12DIV=1./R12
RDIV=1./SQRT(RSQ)
C ----GET A & B VECTOR
C FIRST CHOOSE RANDOM PHI
340 CONTINUE
YY=RGEN(XX)
PHI=YY*COEF12
CPHI=COS(PHI)
SPHI=SIN(PHI)
IF(NCONT.EQ.2)GO TO 11
A1=(-R2*CPHI+R1*R3*RDIV*SPHI)*R12DIV
A2=(R1*CPHI+R2*R3*RDIV*SPHI)*R12DIV
A3=-R12*RDIV*SPHI
C CALL RANDOM PHI
Y2=RGEN(X2)
PHI=Y2*COEF12
CPHI=COS(PHI)
SPHI=SIN(PHI)
B1=(-R2*CPHI+R1*R3*RDIV*SPHI)*R12DIV
B2=(R1*CPHI+R2*R3*RDIV*SPHI)*R12DIV
B3=-R12*RDIV*SPHI
GO TO 12
11 CONTINUE
INDEX=(YY+0.25)*4
PHI=0.7853982*(2*INDEX-1)
A1=SIN(PHI)
A2=COS(PHI)
A3=0.
Y1=RGEN(X1)
INDEX=(Y1+0.25)*4
PHI=0.7853982*(2*INDEX-1)
B1=SIN(PHI)
B2=COS(PHI)
B3=0.
NCONT=1
12 CONTINUE
C DETERMINE A AND B IN EQUATION (4.6)
C RANDOM THETA
Y3=RGEN(X3)

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

THETA=Y3*COEF12
CA=COS(THETA)
CB=SIN(THETA)
UR(II,JJ,LL)=QN*CA*A1
VR(II,JJ,LL)=QN*CA*A2
WR(II,JJ,LL)=QN*CA*A3
UI(II,JJ,LL)=QN*CB*B1
VI(II,JJ,LL)=QN*CB*B2
WI(II,JJ,LL)=QN*CB*B3
IF (N3 .NE. 0) GO TO 20
WSIGN=ABS(A3)/A3
VSIGN=ABS(B3)/B3
WRAN=QNA*CA*RANISO
WIAN=QNA*CB*RANISO
WR(II,JJ,LL)=WR(II,JJ,LL)+WRAN*WSIGN
WI(II,JJ,LL)=WI(II,JJ,LL)+WIAN*VSIGN
20 CONTINUE
30 CONTINUE
40 CONTINUE
C NOW THE UPPER HALF OF THE K-SPACE HAS BEEN DETERMINED
C GET THE TRANSFORMED VELOCITY AT THE CONJUGATE POINTS
C-----CONJUGATE FORM
C N3=1 TO 7, N1 & N2=-7 TO 7
C N3=L-1 -N3=LM
C N2=J-1 -N2=JM
IMP2=IMAX+2
DO 41 L=2,NHALF
LM=L+IMP2-2*L
DO 41 J=1,JMAX
M=(J+IMM1)/IMP1
JM=J+(IMP2-2*J)*M
DO 41 I=1,IMAX
M=(I+IMM1)/IMP1
IM=I+(IMP2-2*I)*M
UR(IM,JM,LM)= UR(I,J,L)
VR(IM,JM,LM)= VR(I,J,L)
WR(IM,JM,LM)= WR(I,J,L)
UI(IM,JM,LM)=-UI(I,J,L)
VI(IM,JM,LM)=-VI(I,J,L)
WI(IM,JM,LM)=-WI(I,J,L)
41 CONTINUE
C N3=0, N1=1 TO 7, N2=-7 TO 7
DO 42 I=2,NHALF
IM=I+(IMP2-2*I)
DO 42 J=1,JMAX
M=(J+IMM1)/IMP1
JM=J+(IMP2-2*J)*M
IF(J.EQ.NHP1) GO TO 42
UR(IM,JM,1 )= UR(I,J,1)
VR(IM,JM,1 )= VR(I,J,1)
WR(IM,JM,1 )= WR(I,J,1)
UI(IM,JM,1 )=-UI(I,J,1)
VI(IM,JM,1 )=-VI(I,J,1)
WI(IM,JM,1 )=-WI(I,J,1)
42 CONTINUE
C N1=N3=0
DO 43 J=2,NHALF
JM=J+(IMP2-2*J)
UR(1,JM,1)= UR(1,J,1)
VR(1,JM,1)= VR(1,J,1)
WR(1,JM,1)= WR(1,J,1)
UI(1,JM,1)=-UI(1,J,1)
VI(1,JM,1)=-VI(1,J,1)
WI(1,JM,1)=-WI(1,J,1)
43 CONTINUE
C X AND Y TRANSFORM
SIGN=-1.
DO 50 L=1,LMAX
CALL MOVLEV(UR(1,1,L),FR(1,1),IJ)
CALL MOVLEV(UI(1,1,L),FI(1,1),IJ)

```

```

CALL FFTX(SIGN)
CALL FFTY(SIGN,CC)
CALL MOVLEV(FR(1,1),UR(1,1,L),IJ)
CALL MOVLEV(FI(1,1),UI(1,1,L),IJ)
CALL MOVLEV(VR(1,1,L),FR(1,1),IJ)
CALL MOVLEV(VI(1,1,L),FI(1,1),IJ)
CALL FFTX(SIGN)
CALL FFTY(SIGN,CC)
CALL MOVLEV(FR(1,1),VR(1,1,L),IJ)
CALL MOVLEV(FI(1,1),VI(1,1,L),IJ)
CALL MOVLEV(WR(1,1,L),FR(1,1),IJ)
CALL MOVLEV(WI(1,1,L),FI(1,1),IJ)
CALL FFTX(SIGN)
CALL FFTY(SIGN,CC)
CALL MOVLEV(FR(1,1),WR(1,1,L),IJ)
CALL MOVLEV(FI(1,1),WI(1,1,L),IJ)
50 CONTINUE
C Z TRANSFORM
DO 51 I=1,IMAX
DO 52 L=1,LMAX
DO 52 J=1,JMAX
FR(J,L)=UR(I,J,L)
FI(J,L)=UI(I,J,L)
52 CONTINUE
CALL FFTY(SIGN,1.0)
DO 53 L=1,LMAX
DO 53 J=1,JMAX
UR(I,J,L)=FR(J,L)
53 CONTINUE
51 CONTINUE
DO 54 I=1,IMAX
DO 55 L=1,LMAX
DO 55 J=1,JMAX
FR(J,L)=VR(I,J,L)
FI(J,L)=VI(I,J,L)
55 CONTINUE
CALL FFTY(SIGN,1.0)
DO 56 L=1,LMAX
DO 56 J=1,JMAX
VR(I,J,L)=FR(J,L)
56 CONTINUE
54 CONTINUE
DO 57 I=1,IMAX
DO 58 L=1,LMAX
DO 58 J=1,JMAX
FR(J,L)=WR(I,J,L)
FI(J,L)=WI(I,J,L)
58 CONTINUE
CALL FFTY(SIGN,1.0)
DO 59 L=1,LMAX
DO 59 J=1,JMAX
WR(I,J,L)=FR(J,L)
59 CONTINUE
57 CONTINUE
C-----THE INITIAL FIELD HAS BEEN GENERATED. THE FOLLOWING IS TO PRINT
C OUT INFORMATION ON THE GENERATED FIELD
C VELOCITIES ARE STORED IN UR, VR AND WR
C-----TURBULENT ENERGY CHECK
TKU=0.
TKV=0.
TKW=0.
DO 95 L=1, LMAX
DO 95 J=1, JMAX
DO 95 I=1, IMAX
TKU=TKU+UR(I,J,L)**2
TKV=TKV+VR(I,J,L)**2
TKW=TKW+WR(I,J,L)**2
95 CONTINUE
TKU=TKU*TDIV
TKV=TKV*TDIV

```

```

TKW=TKW*TDIV
TKSUM=TKU+TKV+TKW
TKUR=TKU/TKSUM
TKVR=TKV/TKSUM
TKWR=TKW/TKSUM
PRINT 707
PRINT 706
PRINT 700,DT,DELTA,C,NAVG,ANISO,UTM,GAMMA
PRINT 702, TKU,TKV,TKW, TKSUM
PRINT 702,TKUR,TKVR,TKWR
PRINT 706
PRINT 601.
UTOT=0.
VTOT=0.
WTOT=0.
DO 120 L=1,LMAX
PRINT 710,L
USUM=0.
VSUM=0.
WSUM=0.
DO 116 J=1,JMAX
DO 116 I=1,IMAX
USUM=USUM+UR(I,J,L)
VSUM=VSUM+VR(I,J,L)
WSUM=WSUM+WR(I,J,L)
116 CONTINUE
PRINT 702,(UR(I,10,L),I=1,NHALF)
PRINT 702,(VR(I,10,L),I=1,NHALF)
PRINT 702,(WR(I,10,L),I=1,NHALF)
PRINT 702, (USUM,VSUM,WSUM)
UTOT=UTOT+USUM
VTOT=VTOT+VSUM
WTOT=WTOT+WSUM
120 CONTINUE
PRINT 702,UTOT,VTOT,WTOT
700 FORMAT(1X,*,INITIAL CONDITION. DT=*1PE10.4,*, DELTA=*1PE10.4,
1 * C=*0PF7.4,3X,*,AVERAGING GRID=*F4.1,*, DELTA*,/,18X,
2*ANISO=*E12.5,3X,*,UTM=*E12.5,3X,*,GAMMA=*E12.5)
702 FORMAT(1P8E15.7)
705 FORMAT(1X,*,CONTINUED AT TIME STEP=*,I4,/,/)
706 FORMAT(1H0,130H*****
-*****
-*****
)
707 FORMAT (1H1)
710 FORMAT(1X,*,PLANE=*,I3)
711 FORMAT(1X,*, INITIAL CONDITION*,/1X,*,DT=*1PE10.4,*, DELTA=*1PE10.4
- ,* C=*F7.4,*, U0=*1PE10.4,/)
601 FORMAT(1X,*,UM,VM,WM*)
RETURN
END
*DECK INIFILT
SUBROUTINE INIFILT(U,N1,N2,N3)
DIMENSION U(N1,N2,N3)
C*****
C THIS SUBROUTINE IS USED TO FILTER THE INITIAL FIELD WITH A WIDE
C FILTER ONLY IN THE Z-DIRECTION.
C*****
*CALL FLT
*CALL DAT21
*CALL DATA7
*CALL DATA9
LEVEL 2,U
LMAXM1=LMAX-1
DO 5 L=1,LMAX
XR(L)=EXP(-FLOAT(L-1)*2/8.0)
5 CONTINUE
AREA=0.5*XR(1)
DO 6 L=2,LMAXM1
AREA=AREA+XR(L)
6 CONTINUE

```



```

      AREA=AREA+0.5*XR(LMAX)
      DO 7 L=1,LMAX
      XR(L)=XR(L)/AREA
7     CONTINUE
      CALL FDCT(1.0)
      DO 8 L=1,LMAX
      FILT3(L)=XR(L)
8     CONTINUE
      DO 1 J=1,JMAX
      DO 1 I=1,IMAX
      DO 2 L=1,LMAX
      XR(L)=U(I,J,L)
2     CONTINUE
      CALL FDCT(1.0)
      DO 3 L=1,LMAX
      XR(L)=XR(L)*FILT3(L)
3     CONTINUE
      CALL FDCT(-1.0)
      DO 4 L=1,LMAX
      U(I,J,L)=XR(L)
4     CONTINUE
1     CONTINUE
      RETURN
      END
*DECK INVERS
      SUBROUTINE INVERS(G,PM,HM,IC,N1,N2,N3)
C*****
C      INVERS IS A POISSON SOLVER. IC = 3 IT EXPANDS THE VARIABLE G      *
C      IN COSINE SERIES IN THE Z-DIRECTION OTHERWISE IT EXPANDS G      *
C      IN SINE SERIES IN THE Z-DIRECTION. IN THE OTHER TWO DIRECTIONS *
C      FOURIER SERIES ARE USED TO EXPAND G                             *
C*****
      DIMENSION G(N1,N2,N3),PM(N1,N2,N3),HM(N1,N2,N3)
*CALL DATA9
*CALL DAT21
*CALL DATA7
*CALL WV
      LEVEL 2,G,PM
      IJ=N1*N2
      CC=1./(IMAX*JMAX)
C*****TRANSFER G TO HM
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      DO 20 L=1,LMAX
      XR(L)=-G(I,J,L)
20     CONTINUE
      IF (IC .EQ. 3) GO TO 100
      CALL FDST(1.0)
      GO TO 200
100    CALL FDCT(1.0)
200    DO 30 L=1,LMAX
      HM(I,J,L)=XR(L)
30     CONTINUE
10     CONTINUE
      DO 40 L=1,LMAX
      CALL MOVLEV(HM(1,1,L),FR(1,1),IJ)
      CALL FFTX(1.0)
      CALL FFTY(1.0,1.0)
      DO 50 J=1,JMAX
      DO 50 I=1,IMAX
      WAV=WAVEXS(I)+WAVEYS(J)+WAVEZS(L)
      IF (ABS(WAV) .LT. 0.00001) GO TO 500
      WAV=1./WAV
      FR(I,J)=FR(I,J)*WAV
      FI(I,J)=FI(I,J)*WAV
      GO TO 50
500    FR(I,J)=0.
      FI(I,J)=0.
50     CONTINUE
      CALL FFTY(-1.0,CC)

```

```

      CALL FFTX(-1.0)
      CALL MOVLEV(FR(1,1),HM(1,1,L),IJ)
40    CONTINUE
      DO 60 J=1,JMAX
      DO 60 I=1,IMAX
      DO 70 L=1,LMAX
      XR(L)=HM(I,J,L)
70    CONTINUE
      IF (IC .EQ. 3) GO TO 300
      CALL FDST(-1.0)
      GO TO 400
300   CALL FDCT(-1.0)
400   DO 80 L=1,LMAX
      PM(I,J,L)=XR(L)
      80 CONTINUE
      60 CONTINUE
      RETURN
      END
*DECK MEANINI
      SUBROUTINE MEANINI
      COMMON/NORM/ DELU,THETA
*CALL DEL
C
      COMMON/LARGE4/01D(16,16,33),02D(16,16,33),03D(16,16,33)
      LEVEL 2,01D,02D,03D
*CALL BLANK
*CALL DIM
*CALL LARGE2
*CALL LARGE3
*CALL LARGE5
*CALL DATA9
C*****
C   THIS ROUTINE CREATES THE MEAN INITIAL FIELD . INITIAL SPICKS   *
C   ARE STORED IN GU THEN FILTERED TO THE CREAT THE GAUSSIAN CORE . *
C*****
      DO 500 L=1,LMAX
      DO 500 J=1,JMAX
      DO 500 I=1,IMAX
      GU(I,J,L)=0.
500   CONTINUE
      DO 501 J=1,JMAX
      GU( 6,J,17)=20.
      GU(11,J,17)=20.
501   CONTINUE
      PRINT 1110
1110  FORMAT(1H1,5X,* INITIAL VORTEX AT PLANE 1*)
      PRINT 1115,(((GU(I,1,L),I=1,IMAX),L),L=1,LMAX)
1115  FORMAT(1X,16F8.2,I3)
      CALL STFIL7
      CALL SFILTER(GU,DUDX,N1,N2,N3)
      PRINT 1113
1113  FORMAT(1H1,5X,*FILTERED VORTEX AT PLANE 1*)
      PRINT 1115,(((GU(I,1,L),I= 1,16 ),L),L=1,LMAX)
      DO 508 L=1,LMAX
      DUDX(1,1,L)=0.0
      DO 508 J=1,JMAX
      DO 508 I=1,IMAX
      DUDX(1,1,L)=DUDX(1,1,L)+GU(I,J,L)/(IMAX*JMAX)
508   CONTINUE
      DO 509 L=1,LMAX
      DO 509 J=1,JMAX
      DO 509 I=1,IMAX
      GU(I,J,L)=DUDX(1,1,L)
509   CONTINUE
      DO 502 L=1,LMAX
      DO 502 J=1,JMAX
      DO 502 I=1,IMAX
      O2(I,J,L)=O2(I,J,L)+GU(I,J,L)
502   CONTINUE
      CALL INVERS(O1,GU,DUDX,1,N1,N2,N3)

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

CALL INVERS(02,GV,DUDX,2,N1,N2,N3)
CALL INVERS(03,GW,DUDX,3,N1,N2,N3)
CALL CURLO(GU,GV,GW,U,V,W,N1,N2,N3)
C COMPUTE THE MEAN INITIAL VELOCITY FIELD AND NORMILIZE THE EQUATION
C WITH DELTA U(DELU) FOR THE VELOCITY SCALE AND THETA THE MOMENTUM
C THICKNESS FOR THE LENGTH SCALES.
CDUM=1./(IMAX*JMAX)
DO 506 L=1,LMAX
DUDX(1,1,L)=0.
DO 506 J=1,JMAX
DO 506 I=1,IMAX
DUDX(1,1,L)=DUDX(1,1,L)+U(I,J,L)*CDUM
506 CONTINUE
DELU=DUDX(1,1,LMAX)-DUDX(1,1,1)
THETA=(0.25-(DUDX(1,1,1)/DELU)**2)*0.5
LMAXM1=LMAX-1
DO 504 L=2,LMAXM1
THETA=THETA+(0.25-(DUDX(1,1,L)/DELU)**2)
504 CONTINUE
THETA=THETA+(0.25-(DUDX(1,1,LMAX)/DELU)**2)*0.5
THETA=THETA*DELTAX
DELTAX=DELTAX/THETA
DELTAY=DELTAY/THETA
DELTAZ=DELTAY/THETA
CALL STWV
DO 505 L=1,LMAX
DO 505 J=1,JMAX
DO 505 I=1,IMAX
U(I,J,L)=U(I,J,L)/DELU
V(I,J,L)=V(I,J,L)/DELU
W(I,J,L)=W(I,J,L)/DELU
505 CONTINUE
CALL CURLU(U,V,W,01,02,03,N1,N2,N3)
PRINT 708
708 FORMAT(1H0,130H*****
1*****
2***** )
DT=0.03125*DELU/THETA
PRINT 1116,DELU,THETA,DT
1116 FORMAT(1X,1H*,* DELU=*,E15.7,10X,* THETA=*,E15.7,10X,* DT=*,E15.7,
&46X,1H*)
PRINT 1117,DELTAX,DELTAY,DELTAX
1117 FORMAT(1X,1H*,* DELTAX=*,E15.7,10X,* DELTAY=*,E15.7,10X,* DELTAX=*
&E15.7,35X,1H*)
PRINT 708
CALL RNDINIC
UDUM=0.
DO 507 L=1,LMAX
DO 507 J=1,JMAX
DO 507 I=1,IMAX
IF(ABS(U(I,J,L)).GT.UDUM) UDUM=ABS(U(I,J,L))
IF(ABS(V(I,J,L)).GT.UDUM) UDUM=ABS(V(I,J,L))
IF(ABS(W(I,J,L)).GT.UDUM) UDUM=ABS(W(I,J,L))
507 CONTINUE
CUDUM=0.30/UDUM
DO 510 L=1,LMAX
DO 510 J=1,JMAX
DO 510 I=1,IMAX
U(I,J,L)=U(I,J,L)*CUDUM
V(I,J,L)=V(I,J,L)*CUDUM
W(I,J,L)=W(I,J,L)*CUDUM
510 CONTINUE
CALL CURLU(U,V,W,01D,02D,03D,N1,N2,N3)
DO 512 L=1,LMAX
DO 512 J=1,JMAX
DO 512 I=1,IMAX
01(I,J,L)=01(I,J,L)+01D(I,J,L)
02(I,J,L)=02(I,J,L)+02D(I,J,L)
03(I,J,L)=03(I,J,L)+03D(I,J,L)
512 CONTINUE

```

```

      CALL INVERS(01, GU, DUDX, 1, N1, N2, N3)
      CALL INVERS(02, GV, DUDX, 2, N1, N2, N3)
      CALL INVERS(03, GW, DUDX, 3, N1, N2, N3)
      CALL CURLO(GU, GV, GW, U, V, W, N1, N2, N3)
      DO 513 L=1, LMAX
      DO 513 J=1, JMAX
      DO 513 I=1, IMAX
      O1D(I, J, L)=0.
      O2D(I, J, L)=0.
      O3D(I, J, L)=0.
513  CONTINUE
      DUMM1=0.
      DUMM2=0.
      DO 3333 I=1, IMAX
      DO 3333 J=1, JMAX
      DO 3333 L=1, LMAX
      IF(O1(I, J, L).GT.DUMM1) DUMM1=O1(I, J, L)
      IF(O2(I, J, L).GT.DUMM2) DUMM2=O2(I, J, L)
3333  CONTINUE
      PRINT 3334, DUMM1, DUMM2
3334  FORMAT(1X, * DUMM1= *, E15.7, 2X, * DUMM2= *, E15.7)
      RETURN
      END
*DECK PARPLOT
      SUBROUTINE PARPLOT
C*****
C      THIS ROUTINE PLOTS THE PARTICLES TRACKS .
C      XMIN IS FIXED TO BE ZERO
C      ZMIN IS FIXED TO BE ZERO
C      XMAX IS FLOATING AND DEPENDS ON NUMBER OF MESHES USED AND DELTAX
C      ZMAX IS FLOATING AND DEPENDS ON NUMBER OF MESHES USED AND DELTAZ
C      SCX IS THE SCALING FACTOR TO ADJUST TO A PAGE LENGHT OF 8 INCHES
C      SCZ IS THE SCALING FACTOR TO ADJUST TO A PAGE LENGHT OF 8 INCHES
C*****
*CALL DATA9
*CALL DEL
*CALL XL
      DATA LB/1HX/
      DATA NL/1HZ/
      XMIN=0.
      XMAX=(IMAX-1)*DELTAX
      ZMIN=0.
      ZMAX=(LMAX-1)*DELTAZ
      PPXMAX=13.0
      PPZMAX=10.
      SCX=PPXMAX/XMAX
      SCZ=PPZMAX/ZMAX
      CALL LINAXS(0., 0., PPXMAX, PPZMAX, .1, -1, 10, 1, XMIN, XMAX, 3, 4, LB)
      CALL LINAXS(0., 0., PPXMAX, PPZMAX, .1, +1, 10, 1, ZMIN, ZMAX, 3, 4, NL)
      DO 1 N=1, 140
      X=XPART(N)*SCX
      Z=ZPART(N)*SCZ
      NC=NCHAR(N)
      CALL SYMBOL(X, Z, 0.1, 0, 0., -NC)
1  CONTINUE
      CALL PLOT(0., 0., 6)
      RETURN
      END
*DECK PARTRAC
      SUBROUTINE PARTRAC(NPART, DT)
C*****
C      THIS SUBROUTINE COMPUTES THE PARTICLES TRACK OF A TWO DIM MEAN.
C      IT USES LINEAR INTERPOLATION TO COMPUTE THE VELOCITIES BETWEEN
C      THE MESHES. TIME ADVANCING IS A FIRST ORDER EULER METHOD.
C*****
*CALL LARGE2
*CALL DATA9
*CALL DEL
*CALL LARGE3
*CALL XL

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

RLX=(IMAX-1)*DELTAX
RLY=(JMAX-1)*DELTAY
RLZ=(LMAX-1)*DELTAZ
DO 1 M=1,NPART
IX=XPART(M)/DELTAX+1
IY=YPART(M)/DELTAY+1
LZ=ZPART(M)/DELTAZ+1
IXP1=IX+1
IYP1=IY+1
LZP1=LZ+1
IF(IX.EQ.IMAX) IXP1=1
IF(IY.EQ.JMAX) IYP1=1
IF(LZ.EQ.LMAX) LZP1=LZ
CCX=(XPART(M)-(IX-1)*DELTAX)/DELTAX
CCY=(YPART(M)-(IY-1)*DELTAY)/DELTAY
CCZ=(ZPART(M)-(LZ-1)*DELTAZ)/DELTAZ
U1PART=U(IX,IY,LZ)+(U(IXP1,IY,LZ)-U(IX,IY,LZ))*CCX
V1PART=V(IX,IY,LZ)+(V(IXP1,IY,LZ)-V(IX,IY,LZ))*CCX
W1PART=W(IX,IY,LZ)+(W(IXP1,IY,LZ)-W(IX,IY,LZ))*CCX
U2PART=U(IX,IY,LZP1)+(U(IXP1,IY,LZP1)-U(IX,IY,LZP1))*CCX
V2PART=V(IX,IY,LZP1)+(V(IXP1,IY,LZP1)-V(IX,IY,LZP1))*CCX
W2PART=W(IX,IY,LZP1)+(W(IXP1,IY,LZP1)-W(IX,IY,LZP1))*CCX
U1PART=U1PART+(U2PART-U1PART)*CCZ
W2PART=W(IX,IY,LZP1)+(W(IXP1,IY,LZP1)-W(IX,IY,LZP1))*CCX
U1PART=U1PART+(U2PART-U1PART)*CCZ
V1PART=V1PART+(V2PART-V1PART)*CCZ
W1PART=W1PART+(W2PART-W1PART)*CCZ
U2PART=U(IX,IYP1,LZ)+(U(IXP1,IYP1,LZ)-U(IX,IYP1,LZ))*CCX
V2PART=V(IX,IYP1,LZ)+(V(IXP1,IYP1,LZ)-V(IX,IYP1,LZ))*CCX
W2PART=W(IX,IYP1,LZ)+(W(IXP1,IYP1,LZ)-W(IX,IYP1,LZ))*CCX
U3PART=U(IX,IYP1,LZP1)+(U(IXP1,IYP1,LZP1)-U(IX,IYP1,LZP1))*CCX
V3PART=V(IX,IYP1,LZP1)+(V(IXP1,IYP1,LZP1)-V(IX,IYP1,LZP1))*CCX
W3PART=W(IX,IYP1,LZP1)+(W(IXP1,IYP1,LZP1)-W(IX,IYP1,LZP1))*CCX
U2PART=U2PART+(U3PART-U2PART)*CCZ
V2PART=V2PART+(V3PART-V2PART)*CCZ
W2PART=W2PART+(W3PART-W2PART)*CCZ
U1PART=U1PART+(U2PART-U1PART)*CCY
V1PART=V1PART+(V2PART-V1PART)*CCY
W1PART=W1PART+(W2PART-W1PART)*CCY
XPART(M)=XPART(M)+DT*U1PART
YPART(M)=YPART(M)+DT*V1PART
ZPART(M)=ZPART(M)+DT*W1PART
IF(XPART(M).GT.RLX) GO TO 10
GO TO 20
10 XPART(M)=XPART(M)-RLX
20 IF(XPART(M).LT.0.) GO TO 30
GO TO 40
30 XPART(M)=XPART(M)+RLX
40 IF(YPART(M).GT.RLY) GO TO 70
GO TO 80
70 YPART(M)=YPART(M)-RLY
80 IF(YPART(M).LT.0.) GO TO 90
GO TO 100
90 YPART(M)=YPART(M)+RLY
100 IF(ZPART(M).GT.RLZ) GO TO 50
GO TO 60
50 ZPART(M)=RLZ
60 IF(ZPART(M).LT.0.) ZPART(M)=0.
1 CONTINUE
RETURN
END
*DECK PARTIAL
SUBROUTINE PARTIAL(M,U,N1,N2,N3)
DIMENSION U(N1,N2,N3)
*CALL DATA9
*CALL BLANK
*CALL WV
*CALL DATA7
LEVEL 2,U

```

```

      IJ=N1*N2
      IF (M.EQ. 2) GO TO 30
C*****DERIVATIVE IN THE X-DIRECTION*****
      DO 10 L=1,LMAX
      CALL MOVLEV(U(1,1,L),FR(1,1),IJ)
      CALL FFTX(1.0)
      DO 15 J=1,JMAX
      DO 15 I=1,IMAX
      DUM=FI(I,J)
      FI(I,J)=WAVEX(I)*FR(I,J)
      FR(I,J)=-WAVEX(I)*DUM
15    CONTINUE
      CALL FFTX(-1.0)
      CALL MOVLEV(FR(1,1),DUDX(1,1,L),IJ)
10    CONTINUE
      GO TO 300
C*****DERIVATIVE IN THE Y- DIRECTION *****
30    CONTINUE
      DO 35 L=1,LMAX
      CALL MOVLEV(U(1,1,L),FR(1,1),IJ)
      DO 32 J=1,JMAX
      DO 32 I=1,IMAX
      FI(I,J)=0.0
32    CONTINUE
      CALL FFTY(1.0,1.0)
      DO 40 J=1,JMAX
      DO 40 I=1,IMAX
      DUM=FI(I,J)
      FI(I,J)=WAVEY(J)*FR(I,J)
      FR(I,J)=-WAVEY(J)*DUM
40    CONTINUE
      CALL FFTY(-1.0,1.0)
      CALL MOVLEV(FR(1,1),DUDX(1,1,L),IJ)
35    CONTINUE
300   CONTINUE
      RETURN
      END
*DECK RGEN
      IDENT RGEN - PSEUDO RANDOM NUMBER GENERATOR
***    FUNCTION RGEN(D)
*      CALLED AS A FUNCTION WITH 1 ARGUMENT (WHICH IS IGNORED)
*      RETURNS IN X6 A RANDOM NUMBER GENERATED BY MULTIPLYING
*      1 OF 5 INTEGER CONSTANTS BY THE CORRESPONDING GENERATOR
*      SEE BKY USERS HANDBOOK FOR REFERENCES
      SST
**    RGENCOM - USED TO STORE THE GENERATORS AND POINTER
*
      USE /RGENCOM/
      GEN DATA 1048015011D THE 5 GENERATORS
      DATA 2236846573D
      DATA 4216793093D
      DATA 7792106907D
      DATA 9630191977D
      PTR DATA 1 POINTER TO CURRENT GENERATOR
      USE *
      ENTRY RGEN
      IF -DEF,FTN,1
      ENTRY RGEN$
      COMMENT RANDOM NUMBER GENERATOR (#MODLEVEL#)
      NAME VFD 42/4LRGEN,18/RGEN
      IF DEF,FTN
      ELSE 2
      RGEN PS ENTRY / EXIT
      RGEN$ EQU RGEN SINCE ARG IS IGNORED
      SA1 PTR GET POINTER
      SA3 RGEN GET ENTRY POINT
      SB1 1
      SX7 4
      SA4 X1+GEN-1 GET GENERATOR
      SA5 X1+CON-1 GET CONSTANT

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

      MX2      60-35
      AX3      30
      IX0      X7-X1      4 - PTR
      SB7      X3          B7 = RETURN ADDRESS
      DX6      X4*X5      GEN * CON
      PL       X0,RGEN1    UNLESS PTR =5
      MX1      0          IF PTR WAS 5
RGEN1  BSS      0
      SA3      EXP
      SX7      B1+X1      INCREMENT PTR
      SA7      PTR        STORE NEW POINTER
      BX7      -X2*X6      MASK LOW 35 BITS
      BX5      X7+X3      PUT IN EXPONENT
      SA7      A4          STORE NEW GENERATOR
      NX6      X5
      JPB      B7          JUMP DIRECTLY BACK
CON     DATA   131075D      CONSTANTS TO MULTIPLY BY
      DATA   163843D
      DATA   196611D
      DATA   229379D
      DATA   262147D
EXP     DATA   1734BS48
      END
*DECK RNDINIC
      SUBROUTINE RNDINIC
C*****
C      THIS SUBROUTINE CREATES THE INITIAL RANDOM FIELD BY CALL INICON *
C      INICON WAS INITIALLY WRITTEN BY KWAK,D. AND IS USED HERE AS A MEAN *
C      TO CREATE A RANDOM INITIAL FIELD.THE ROUTINE WAS WRITTEN FOR EQUAL *
C      MESH,AND HENCE THE COMPLICATIONS IN THIS ROUTINE TO TRANSFER THE *
C      FIELD TO THE MIDDLE OF THE BOX .
C*****
*CALL BLANK
*CALL LARGE2
*CALL LARGE3
      COMMON/LARGE4/01(16,16,33),02(16,16,33),03(16,16,33)
      LEVEL 2,01,02,03
*CALL DATA9
*CALL DIM
      COMMON/DUM1/ UM(16,16,16),VM(16,16,16),WM(16,16,16)
      COMMON/DUM2/ GUI(16,16,16),GVI(16,16,16),GWI(16,16,16)
      LEVEL 2,UM,VM,WM,GUI,GVI,GWI
      CALL INICON(C,COF,DT,UM,VM,WM,GUI,GVI,GWI,16,16,16)
      IMAX=N1
      JMAX=N2
      LMAX=N3
      DO 1 L=1,16
      DO 1 J=1,JMAX
      DO 1 I=1,IMAX
      L1=25-L
      L2=17-L
      U(I,J,L1)=UM(I,J,L2)
      V(I,J,L1)=VM(I,J,L2)
      W(I,J,L1)=WM(I,J,L2)
1 CONTINUE
      DO 2 L=1,13
      DO 2 J=1,JMAX
      DO 2 I=1,IMAX
      U(I,J,L)=0.
      V(I,J,L)=0.
      W(I,J,L)=0.
2 CONTINUE
      DO 3 L=21,LMAX
      DO 3 J=1,JMAX
      DO 3 I=1,IMAX
      U(I,J,L)=0.
      V(I,J,L)=0.
      W(I,J,L)=0.
3 CONTINUE
      CALL INIFILT(U,N1,N2,N3)

```

```

      CALL INIFILT(V,N1,N2,N3)
      CALL INIFILT(W,N1,N2,N3)
      CALL CURLO(U,V,W,01,02,03,N1,N2,N3)
      DO 16 L=1,LMAX
      DO 16 J=1,JMAX
      DO 16 I=1,IMAX
      U(I,J,L)=01(I,J,L)
      V(I,J,L)=02(I,J,L)
      W(I,J,L)=03(I,J,L)
16  CONTINUE
      CALL CURLU(U,V,W,01,02,03,N1,N2,N3)
      CALL INVERS(01,GU,DUDX,1,N1,N2,N3)
      CALL INVERS(02,GV,DUDX,2,N1,N2,N3)
      CALL INVERS(03,GW,DUDX,3,N1,N2,N3)
      CALL CURLO(GU,GV,GW,U,V,W,N1,N2,N3)
      RETURN
      END
*DECK SINPART
      SUBROUTINE SINPART(U,N1,N2,N3)
C*****
C      THIS ROUTINE COMPUTES THE PARTIAL DERIVATIVE OF U IN THE Z-
C      DIRECTION BY EXPANDING IN FOURIER SINE SERIES.
C      THE PARTIAL IS STORED IN DUDX.
C*****
      DIMENSION U(N1,N2,N3)
*CALL BLANK
*CALL WV
*CALL DAT21
*CALL DATA9
      LEVEL 2,U
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      DO 20 L=1,LMAX
      XR(L)=U(I,J,L)
20  CONTINUE
      SIGN=1.0
      CALL FDST(SIGN)
      DO 30 L=1,LMAX
      XR(L)=XR(L)*WAVEZ(L)
30  CONTINUE
      SIGN=-1.0
      CALL FDCT(SIGN)
      DO 40 L=1,LMAX
      DUDX(I,J,L)=XR(L)
40  CONTINUE
10  CONTINUE
      RETURN
      END
*DECK SFILTER
      SUBROUTINE SFILTER(HR,HI,N1,N2,N3)
C*****
C      SFILTER FILTERS HR BY EXPANDING IT IN A FOURIER SINE SERIES IN
C      THE Z/DIRECTION AND FOURIER SERIES IN THE OTHER TWO DIRECTIONS.
C*****
      DIMENSION HR(N1,N2,N3),HI(N1,N2,N3)
*CALL FLT
*CALL DATA9
*CALL DATA7
*CALL DAT21
      LEVEL 2,HR
      CC=1.0/(IMAX*JMAX)
      IJ=N1*N2
      DO 10 J=1,JMAX
      DO 10 I=1,IMAX
      DO 20 L=1,LMAX
      XR(L)=HR(I,J,L)
20  CONTINUE
      CALL FDST(1.0)
      DO 30 L=1,LMAX
      HI(I,J,L)=XR(L)

```



```

30 CONTINUE
10 CONTINUE
  DO 40 L=1,LMAX
    CALL MOVLEV(HI(1,1,L),FR(1,1),IJ)
    CALL FFTX(1.0)
    CALL FFTY(1.0,1.0)
    DO 50 J=1,JMAX
      DO 50 I=1,IMAX
        FR(I,J)=FR(I,J)*FILT1(I)*FILT2(J)*FILT3(L)
        FI(I,J)=FI(I,J)*FILT1(I)*FILT2(J)*FILT3(L)
50 CONTINUE
    CALL FFTX(-1.0)
    CALL FFTY(-1.0,CC)
    CALL MOVLEV(FR(1,1),HI(1,1,L),IJ)
40 CONTINUE
  DO 60 J=1,JMAX
    DO 60 I=1,IMAX
      DO 70 L=1,LMAX
        XR(L)=HI(I,J,L)
70 CONTINUE
    CALL FDST(-1.0)
    DO 80 L=1,LMAX
      HR(I,J,L)=XR(L)
80 CONTINUE
60 CONTINUE
  RETURN
  END
*DECK SGS
  SUBROUTINE SGS(U,V,E,N1,N2,N3)
    DIMENSION U(N1,N2,N3),V(N1,N2,N3),E(N1,N2,N3)
*CALL DEL
*CALL LARGE3
*CALL LARGE5
*CALL BLANK
*CALL DATA9
    LEVEL 2,U,V,E
C*****
C    THE SGS MODEL IS COMPUTED IN THIS ROUTINE BY SECOND ORDER DIFF *
C    AND STORED IN GU,GV,GW *
C*****
    CSGSX=1./(2.*DELTA X)
    CSGSY=1./(2.*DELTA Y)
    CSGSZ=1./(2.*DELTA Z)
    IJK=N1*N2*N3
    CALL MOVLEV(DUDX(1,1,1),E(1,1,1),IJK)
    DO 210 L=1,LMAX
      LM1=L-1
      LP1=L+1
      IF (L.EQ. 1) LM1=LP1
      IF (L.EQ. LMAX) LP1=LM1
      DO 210 J=1,JMAX
        CALL FIX(JM1,J,JP1,JMAX)
        DO 210 I=1,IMAX
          CALL FIX(IM1,I,IP1,IMAX)
          U(I,J,L)=(E(I,JP1,L)*01(I,JP1,L)-E(I,JM1,L)*01(I,JM1,L))*CSGSY
          V(I,J,L)=(E(I,J,LP1)*01(I,J,LP1)-E(I,J,LM1)*01(I,J,LM1))*CSGSZ
          W(I,J,L)=(E(IP1,J,L)*03(IP1,J,L)-E(IM1,J,L)*03(IM1,J,L))*CSGSX
210 CONTINUE
        CALL PARTIAL(2,U,N1,N2,N3)
        CALL MOVLEV(DUDX(1,1,1),U(1,1,1),IJK)
        CALL COSPART(V,N1,N2,N3)
        DO 220 L=1,LMAX
          DO 220 J=1,JMAX
            DO 220 I=1,IMAX
              GU(I,J,L)=GU(I,J,L)+U(I,J,L)+DUDX(I,J,L)
220 CONTINUE
        DO 230 L=1,LMAX
          LM1=L-1
          LP1=L+1

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR



```

      IF (L .EQ. 1) LM1=LP1
      IF (L .EQ. LMAX) LP1=LM1
      DO 230 J=1,JMAX
      CALL FIX(JM1,J,JP1,JMAX)
      DO 230 I=1,IMAX
      CALL FIX(IM1,I,IP1,IMAX)
      U(I,J,L)=(E(IP1,J,L)*O2(IP1,J,L)-E(IM1,J,L)*O2(IM1,J,L))*CSGSX
      1 -(E(I,JP1,L)*O1(I,JP1,L)-E(I,JM1,L)*O1(I,JM1,L))*CSGSY
      V(I,J,L)=(E(I,J,LP1)*O2(I,J,LP1)-E(I,J,LM1)*O2(I,J,LM1))*CSGSZ
      1 -(E(I,JP1,L)*O3(I,JP1,L)-E(I,JM1,L)*O3(I,JM1,L))*CSGSY
230  CONTINUE
      CALL PARTIAL(1,U,N1,N2,N3)
      CALL MOVLEV(DUDX(1,1,1),U(1,1,1),IJK)
      CALL COSPART(V,N1,N2,N3)
      DO 240 L=1,LMAX
      DO 240 J=1,JMAX
      DO 240 I=1,IMAX
      GV(I,J,L)=GV(I,J,L)+U(I,J,L)+DUDX(I,J,L)
240  CONTINUE
      DO 250 L=1,LMAX
      LM1=L-1
      LP1=L+1
      IF (L .EQ. 1) LM1=LP1
      IF (L .EQ. LMAX) LP1=LM1
      DO 250 J=1,JMAX
      CALL FIX(JM1,J,JP1,JMAX)
      DO 250 I=1,IMAX
      CALL FIX(IM1,I,IP1,IMAX)
      U(I,J,L)=(E(IP1,J,L)*O3(IP1,J,L)-E(IM1,J,L)*O3(IM1,J,L))*CSGSX
      1 -(E(I,J,LP1)*O1(I,J,LP1)-E(I,J,LM1)*O1(I,J,LM1))*CSGSZ
      V(I,J,L)=(E(I,JP1,L)*O3(I,JP1,L)-E(I,JM1,L)*O3(I,JM1,L))*CSGSY
      1 -(E(I,J,LP1)*O2(I,J,LP1)-E(I,J,LM1)*O2(I,J,LM1))*CSGSZ
250  CONTINUE
      CALL PARTIAL(1,U,N1,N2,N3)
      CALL MOVLEV(DUDX(1,1,1),U(1,1,1),IJK)
      CALL PARTIAL(2,V,N1,N2,N3)
      DO 260 L=1,LMAX
      DO 260 J=1,JMAX
      DO 260 I=1,IMAX
      GW(I,J,L)=GW(I,J,L)+U(I,J,L)+DUDX(I,J,L)
260  CONTINUE
      RETURN
      END
*DECK STFLT
      SUBROUTINE STFLT
C*****
C      THIS SUBROUTINE INITIALIZE THE TRANSFORM OF THE FILTER IN EACH
C      DIRECTION. THE TRANSFORM IS STORED IN FILT1,FILT2,FILT3, FOR USE
C      IN SUBROUTINE FILTER.
C*****
*CALL AVG
*CALL FLT
*CALL DATA7
*CALL DAT21
*CALL DATA9
*CALL PR
      NHP1X=IMAX/2+1
      NHP1Y=JMAX/2+1
      NHP2X=NHP1X+1
      NHP2Y=NHP1Y+1
      LMAXM1=LMAX-1
      IF(CCF .NE. 0.) GO TO 400
C*****FIX THE TRANSFORM OF THE FILTER IN THE X-DIRECTION
      DO 100 J=1,JMAX
      DO 100 I=1,NHP1X
      FR(I,J)=EXP(-6.*(FLOAT(I-1)/AVG1)**2)
100  CONTINUE
      DO 110 J=1,JMAX
      DO 110 I=NHP2X,IMAX
      II=IMAX-I+2

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

      FR(I,J)=FR(II,J)
110 CONTINUE
      AREA=0.0
      DO 120 I=1,IMAX
        AREA=AREA+FR(I,1)
120 CONTINUE
      DO 130 J=1,JMAX
        DO 130 I=1,IMAX
          FR(I,J)=FR(I,J)/AREA
          FI(I,J)=0.0
130 CONTINUE
      CALL FFTX(1.0)
      DO 140 I=1,IMAX
        FILT1(I)=FR(I,1)
140 CONTINUE
C*****FIX THE TRANSFORM OF THE FILTER IN THE Y-DIRECTION
      DO 200 J=1,NHPIY
        DO 200 I=1,IMAX
          FR(I,J)=EXP(-6.*(FLOAT(J-1)/AVG2)**2)
200 CONTINUE
      DO 210 J=NHP2Y,JMAX
        DO 210 I=1,IMAX
          JJ=JMAX-J+2
          FR(I,J)=FR(I,JJ)
210 CONTINUE
      AREA=0.0
      DO 220 J=1,JMAX
        AREA=AREA+FR(1,J)
220 CONTINUE
      DO 230 J=1,JMAX
        DO 230 I=1,IMAX
          FR(I,J)=FR(I,J)/AREA
          FI(I,J)=0.0
230 CONTINUE
      CALL FFTY(1.0,1.0)
      DO 240 J=1,JMAX
        FILT2(J)=FR(1,J)
240 CONTINUE
C*****FIX THE TRANSFORM OF THE FILTER IN THE Z-DIRECTION
      DO 300 L=1,LMAX
        XR(L)=EXP(-6.*(FLOAT(L-1)/AVG3)**2)
300 CONTINUE
      AREA=0.5*XR(1)
      DO 310 L=2,LMAXM1
        AREA=AREA+XR(L)
310 CONTINUE
      AREA=AREA+0.5*XR(LMAX)
      DO 320 L=1,LMAX
        XR(L)=XR(L)/AREA
320 CONTINUE
      CALL FDCT(1.0)
      DO 330 L=1,LMAX
        FILT3(L)=XR(L)
330 CONTINUE
      FILT1(NHPIX)=0.
      FILT2(NHPIY)=0.
      FILT3(LMAX)=0.
      GO TO 410
400 IF(CCF.NE.1.0) GO TO 410
      MC=(LMAX-1)*2/3
      DO 7 L=1,LMAX
        FILT3(L)=0.
7 CONTINUE
      DO 8 L=1,MC
        FILT3(L)=1.0
8 CONTINUE
      MC=JMAX/3+1
      DO 9 J=1,JMAX
        FILT2(J)=0.
9 CONTINUE

```

```

DO 10 J=1,MC
FILT2(J)=1.0
JJ=JMAX-J+1
FILT2(JJ)=1.0
10 CONTINUE
MC=JMAX-MC+1
FILT2(MC)=0.
MC=IMAX/3+1
DO 11 I=1,IMAX
FILT1(I)=0.
11 CONTINUE
DO 12 I=1,MC
FILT1(I)=1.0
II=IMAX+1-I
FILT1(II)=1.0
12 CONTINUE
MC=IMAX-MC+1
FILT1(MC)=0.
410 PRINT 1119,CCF
1119 FORMAT(1H0,* CCF=*,1PE15.7)
IF(CCPF .NE. 1.0) GO TO 340
PRINT 1116,(FILT1(I),I=1,IMAX)
PRINT 1117,(FILT2(J),J=1,JMAX)
PRINT 1118,(FILT3(L),L=1,LMAX)
1116 FORMAT(1X,* FILT1 =*,1PE15.7)
1117 FORMAT(1X,* FILT2 =*,1PE15.7)
1118 FORMAT(1X,* FILT3 =*,1PE15.7)
340 CONTINUE
RETURN
END
*DECK STPART
SUBROUTINE STPART(NPART)
*CALL LARGE5
*CALL XL
*CALL DATA9
*CALL DEL
YPART(1)=0.
YPART(17)=0.
XPART(1)=(6-1)*DELTAX
XPART(17)=(11-1)*DELTAX
ZPART(1)=(17.-1.)*DELTAZ
ZPART(17)=(17.-1.)*DELTAZ
NCHAR(1)=1
NCHAR(17)=2
YPART(33)=0.
YPART(49)=0.
YPART(65)=0.
YPART(81)=0.
YPART(97)=0.
YPART(113)=0.
YPART(129)=0.
YPART(145)=0.
XPART(33)=XPART(1)+0.5*DELTAX
XPART(49)=XPART(1)-0.5*DELTAX
XPART(65)=XPART(1)
XPART(81)=XPART(1)
ZPART(33)=ZPART(1)
ZPART(49)=ZPART(1)
ZPART(65)=ZPART(1)+0.5*DELTAZ
ZPART(81)=ZPART(1)-0.5*DELTAZ
XPART(97)=XPART(17)+0.5*DELTAX
ZPART(97)=ZPART(17)
XPART(113)=XPART(17)-0.5*DELTAX
ZPART(113)=ZPART(17)
XPART(129)=XPART(17)
ZPART(129)=ZPART(17)+0.5*DELTAZ
XPART(145)=XPART(17)
ZPART(145)=ZPART(17)-0.5*DELTAZ
NCHAR( 33)=3
NCHAR( 49)=4

```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```

NCHAR( 65)=5
NCHAR( 81)=6
NCHAR( 97)=7
NCHAR(113)=8
NCHAR(129)=9
NCHAR(145)=10
NPART=160
DELX=0.
DELZ=0.
N=0
DO 1 M=1,10
N=N+1
DO 2 J=2,JMAX
N=N+1
NCHAR(N)=M
IX=XPART(N-1)/DELTAX+1
LZ=ZPART(N-1)/DELTAZ+1
IXP1=IX+1
LZP1=LZ+1
CCX=(XPART(N-1)-(IX-1)*DELTAX)/DELTAX
CCZ=(ZPART(N-1)-(LZ-1)*DELTAZ)/DELTAZ
O1P1=O1(IX,J,LZ)+(O1(IXP1,J,LZ)-O1(IX,J,LZ))*CCX
O2P1=O2(IX,J,LZ)+(O2(IXP1,J,LZ)-O2(IX,J,LZ))*CCX
O3P1=O3(IX,J,LZ)+(O3(IXP1,J,LZ)-O3(IX,J,LZ))*CCX
O1P2=O1(IX,J,LZP1)+(O1(IXP1,J,LZP1)-O1(IX,J,LZP1))*CCX
O2P2=O2(IX,J,LZP1)+(O2(IXP1,J,LZP1)-O2(IX,J,LZP1))*CCX
O3P2=O3(IX,J,LZP1)+(O3(IXP1,J,LZP1)-O3(IX,J,LZP1))*CCX
O1P1=O1P1+(O1P1-O1P2)*CCZ
O2P1=O2P1+(O2P1-O2P2)*CCZ
O3P1=O3P1+(O3P1-O3P2)*CCZ
DELX=O1P1*DELTAY/O2P1
DELZ=O3P1*DELTAY/O2P1
XPART(N)=XPART(N-1)+DELX
YPART(N)=YPART(N-1)+DELTAY
ZPART(N)=ZPART(N-1)+DELZ
2 CONTINUE
1 CONTINUE
RETURN
END
*DECK STREAD
SUBROUTINE STREAD
C*****
C THIS SUBROUTINE READ THE INPUT PARAMETERS *
C IMAX=NUMBER OF GRID POINTS IN THE X-DIRECTION *
C JMAX=NUMBER OF GRID POINTS IN THE Y-DIRECTION *
C LMAX=NUMBER OF GRID POINTS IN THE Z-DIRECTION *
C AVG2=FILTERING WIDTH IN THE Y-DIRECTION *
C AVG3=FILTERING WIDTH IN THE Z-DIRECTION *
C AVG1=FILTERING WIDTH IN THE X-DIRECTION *
C DELTAX= MESH SIZE IN THE X-DIRECTION *
C DELTAY= MESH SIZE IN THE Y-DIRECTION *
C DELTAZ= MESH SIZE IN THE Z-DIRECTION *
C N1= ARRAY SIZE IN THE X-DIRECTION *
C N2= ARRAY SIZE IN THE Y-DIRECTION *
C N3= ARRAY SIZE IN THE Z-DIRECTION *
C CCPW= 1 IF PRINT OUT OF WAVE IS WANTED,OTHERWISE NO PRINT OUT *
C CCPF= 1 IF PRINT OUT OF FILT IS WANTED,OTHERWISE NO PRINT OUT *
C CCPD= 1 IF PRINT OUT OF LINE AVERAGE OF U-COMPONENT *
C LINE AVERAGE OF W-COMPONENT AND ENSEMBLE AVERAGE PERTURBATIONS*
C IS REQUIRED ,OTHERWISE NO PRINT OUT . *
C*****
C INTEGER TSTART,TEND
*CALL DATA9
COMMON/TIM/ TSTART,TEND
*CALL DEL
*CALL DIM
*CALL AVG
*CALL PR
READ 703,IMAX,JMAX,LMAX,TSTART,TEND
READ 704,DELTAX,DELTAY,DELTAZ

```

```

READ 704,AVG1,AVG2,AVG3,CCF
READ 703,N1,N2,N3
READ 704,CCPW,CCPF,CCPD
PRINT 708
PRINT 705,IMAX,JMAX,LMAX,TSTART,TEND
PRINT 706,DELTAX,DELTAY,DELTAZ
PRINT 707,AVG1,AVG2,AVG3
PRINT 709,N1,N2,N3
PRINT 708
703 FORMAT(10I5)
704 FORMAT(4E10.4)
705 FORMAT(1X,* IMAX=*,I5,5X,* JMAX=*,I5,5X,* LMAX=*,I5,5X,* TSART=*,I5
+ ,5X,* TEND=*,I5,52X,1H*)
706 FORMAT(1X,* DELTAX=*,1PE10.4,5X,* DELTAY=*,1 E10.4,5X,* DELTAZ=*,1
+PE10.4,64X,1H*)
707 FORMAT(1X,* AVG1X=*,1PE10.4,5X,* AVG2=*,1 E10.4,5X,* AVG3=*,1
+PE10.4,64X,1H*)
708 FORMAT(1H0,130H*****
1*****
2***** )
709 FORMAT(1X,* N1=*,I5,5X,* N2=*,I5,5X,* N3=*,I5,5X,70X,1H*)
RETURN
END
*DECK STWV
SUBROUTINE STWV
C*****
C STWV SETS THE WAVE NUMBERS FOR A GIVEN MESH SIZE DELTA AND *
C NUMBER OF MESH POINTS NMAX . THIS ROUTINE MUST BE CALLED *
C TO INITIALIZE THE WAVE NUMBERS FOR THE PARTIAL ROUTINES AND *
C INVERS ROUTINE . *
C*****
C*****
*CALL WV
*CALL DATA9
*CALL DEL
*CALL PR
PAI=3.1415926535898
CX=2.0*PAI/(FLOAT(IMAX)*DELTAX)
CY=2.0*PAI/(FLOAT(JMAX)*DELTAY)
CZ=PAI/(FLOAT(LMAX-1)*DELTAZ)
C2X=CX/FLOAT(IMAX)
C2Y=CY/FLOAT(JMAX)
NHPIX=IMAX/2+1
NHPIY=JMAX/2+1
DO 100 L=1,LMAX
WAVEZ(L)=CZ*FLOAT(L-1)
WAVEZS(L)=-WAVEZ(L)**2
100 CONTINUE
DO 101 J=1,JMAX
MM=J/NHPIY
M=MM*JMAX+1
WAVEY(J)=C2Y*FLOAT(J-M)
WAVEYS(J)=-(CY*FLOAT(J-M))**2
101 CONTINUE
DO 102 I=1,IMAX
MM=I/NHPIX
M=MM*IMAX+1
WAVEX(I)=C2X*FLOAT(I-M)
WAVEXS(I)=-(CX*FLOAT(I-M))**2
102 CONTINUE
WAVEX(NHPIX)=0.
WAVEY(NHPIY)=0.
WAVEXS(NHPIX)=0.
WAVEYS(NHPIY)=0.
WAVEZ(LMAX)=0.
WAVEZS(LMAX)=0.
IF(CCPW.NE.1) GO TO 104
PRINT 1000,(WAVEX(I),WAVEXS(I),I=1,IMAX)
PRINT 1001,(WAVEY(J),WAVEYS(J),J=1,JMAX)
PRINT 1002,(WAVEZ(L),WAVEZS(L),L=1,LMAX)

```

```
104 CONTINUE
1000 FORMAT(1X,* WAVEX =*,1PE15.7,5X,* WAVEXS =*,1PE15.7)
1001 FORMAT(1X,* WAVEY =*,1PE15.7,5X,* WAVEYS =*,1PE15.7)
1002 FORMAT(1X,* WAVEZ =*,1PE15.7,5X,* WAVEZS =*,1PE15.7)
      RETURN
      END
```